



InformaticaTM

Accelerators

Informatica MDM - Product 360

Version: 10.1

Table of Contents

1	ECCnet Accelerator	13
1.1	Introduction	13
1.2	Communicating with ECCnet	14
1.2.1	Features of the Accelerator	14
1.3	ECCnet Accelerator Features	14
1.4	ECCnet Accelerator package content	15
1.5	ECCnet Accelerator installation	15
1.5.1	Prerequisites	15
1.5.1.1	Product 360 environment	15
1.5.2	Installing the ECCnet Accelerator	15
1.5.2.1	Download ECCnet Accelerator zip	15
1.5.2.2	Extract the ECCnet Accelerator zip	15
1.5.2.3	Product 360 environment configuration	16
1.5.3	Application modules for ECCnet	17
1.5.4	Configure ECCnet access from Product 360 - Supplier Portal	18
1.5.4.1	Configure Product 360 - Supplier Portal	18
1.5.4.2	Configure Product 360 - Supplier Portal ECCnet System User	19
1.5.4.3	Configure Product 360 - Supplier Portal ECCnet System User within Product 360 - Web	21
1.6	ECCnet Accelerator operation	21
1.6.1	Desktop client	21
1.6.1.1	Access ECCnet search perspective	22
1.6.1.2	Search for items in ECCnet Item Centre	23
1.6.1.3	Browse through the search results	24
1.6.1.4	View packaging hierarchy for items as defined in ECCnet Item Centre	24
1.6.1.5	Subscribe selected items	25
1.6.1.6	Subscribe all the items in the search results from ECCnet Item Centre	27
1.6.1.7	Retrieve (or update) latest data from ECCnet Item Centre for subscribed items	28
1.6.1.8	Unsubscribe items	29
1.6.1.9	View and edit ECCnet items data in Product 360 supplier catalogs	30
1.6.1.10	View subscribed items under ECCnet and GPC classification systems	32
1.6.1.11	View packaging hierarchy for subscribed items in Product 360	32
1.6.1.12	Schedule job to update data for all ECCnet items in a Product 360 catalog	33

1.6.1.13	Schedule job to update data for all ECCnet items in all Product 360 catalogs.....	35
1.6.1.14	View job status and process logs for all recent subscribe/unsubscribe/update data requests (one-off and scheduled)	35
1.6.2	Web client.....	36
1.6.2.1	Access ECCnet	36
1.6.2.2	Search for items in ECCnet Item Centre.....	37
1.6.2.3	Browse through the search results	38
1.6.2.4	View packaging hierarchy for items as defined in ECCnet Item Centre	39
1.6.2.5	Subscribe selected items.....	39
1.6.2.6	Subscribe all the items in the search results	40
1.6.2.7	Retrieve (or update) latest data from ECCnet Item Centre for subscribed items.....	41
1.6.2.8	Unsubscribe items	42
1.6.2.9	View subscribed items in Product 360 under ECCnet and GPC classification systems	43
1.6.2.10	View and edit ECCnet items data in Product 360	44
1.6.2.11	View packaging hierarchy for subscribed items in Product 360.....	45
1.6.3	Supplier Portal	46
1.6.3.1	Configure ECCnet access for a supplier	46
1.6.3.2	Access ECCnet	47
1.7	ECCnet Accelerator field list	47
1.8	ECCnet Accelerator limitations	47
1.8.1	Service API	47
2	GDSN Accelerator	48
2.1	GDSN Accelerator (Major Release 3)	48
2.1.1	Providing an interface to the Global Data Synchronization Network (GDSN) for accurate product data exchange	48
2.1.1.1	There are many benefits of using the GDSN Accelerator:	49
2.1.1.2	Item Management	49
2.1.1.3	Data Sync Engine	49
2.1.2	The GDSN accelerator for Product 360	50
2.1.2.1	Data model extension	50
2.1.2.2	New UI components.....	51
2.1.2.3	Communicating with the GDSN data pool using Informatica B2B DX.....	53
2.1.2.4	Media assets	53
2.1.3	GDSN Accelerator package content	54
2.1.3.1	Selecting the right package	54

2.1.4	GDSN Accelerator installation	55
2.1.4.1	Prerequisites	55
2.1.4.2	Installing the GDSN Accelerator	56
2.1.4.3	Application modules.....	59
2.1.4.4	B2B Data Exchange	60
2.1.4.5	Repository configurations	64
2.1.5	GDSN Accelerator setup.....	69
2.1.5.1	Export templates.....	69
2.1.5.2	Setup automated jobs	87
2.1.6	GDSN Accelerator operation	94
2.1.6.1	Maintain valid data	94
2.1.6.2	Send data.....	96
2.1.6.3	Receive feedback	99
2.1.6.4	Publication	101
2.1.7	GDSN Accelerator field list.....	103
2.1.8	GDSN Accelerator FAQ	216
2.1.8.1	General	216
2.1.8.2	Export	218
2.1.8.3	B2B Data Exchange	219
2.1.9	GDSN implementation guidelines.....	221
2.1.9.1	Introduction	221
2.1.9.2	Documents	222
2.1.9.3	Analyze requirements	222
2.1.9.4	Data model	235
2.1.9.5	UI adjustments	256
2.1.9.6	Data validations	265
2.1.9.7	Communication	270
2.1.9.8	Testing	279
2.1.9.9	FAQ.....	284
2.1.10	GDSN Customization.....	284
2.1.10.1	Customize how items are found for the set publication status rest call.....	284
2.1.11	GDSN 3.1.8 -> GDSN 3.1.9.....	286
2.1.11.1	3.1.8 → 3.1.9.....	286
2.1.11.2	Results:	287
2.1.11.3	Fields:.....	287

2.1.11.4 Validations:.....	287
2.1.11.5 Code lists:	287
2.1.11.6 Units.....	291
2.1.12 GDSN 3.1.9 -> GDSN 3.1.10.....	291
2.1.12.1 3.1.9 → 3.1.10	291
2.1.12.2 Results:	292
2.1.12.3 Fields:.....	292
2.1.12.4 Validations:.....	292
2.1.12.5 Code lists:	292
2.1.13 GDSN 3.1.10 -> GDSN 3.1.11.....	294
2.1.13.1 3.1.10 → 3.1.11	294
2.1.13.2 Results:	294
2.1.13.3 Fields:.....	294
2.1.13.4 Validations:.....	294
2.1.13.5 Code lists:	295
2.1.14 GDSN 3.1.11 -> GDSN 3.1.12.....	296
2.1.14.1 3.1.11 → 3.1.12	296
2.1.14.2 Results:	296
2.1.14.3 Fields:.....	296
2.1.14.4 Validations:.....	297
2.1.14.5 Code lists:	297
2.1.15 Migration Guide for IM	298
2.1.15.1 General changes.....	298
2.1.15.2 GDSN Core Attributes.....	299
2.1.15.3 Food and Beverage	303
2.1.15.4 Merge of migration Data Quality configurations	307
2.1.15.5 Missing Data Quality rules	308
2.1.16 GDSN Migration Guide for version 3.1.3.....	309
2.1.16.1 Overview	309
2.1.16.2 Prerequisite for the Migration	310
2.1.16.3 Migration	312
2.1.16.4 Product 360 data model changes.....	316
2.1.16.5 Data quality rule and configuration changes	319
2.1.16.6 Additional changes	322
2.1.16.7 Valid value list changes.....	325

2.1.16.8 DSE Export template changes	333
2.1.16.9 IM Export template changes.....	341
2.1.17 GDSN Migration Guide for version 3.1.12	348
2.1.17.1 Overview	348
2.1.17.2 Product 360 data model changes.....	349
2.1.17.3 Data quality rule and configuration changes	350
2.1.17.4 Changes in data source export templates	353
2.1.17.5 B2B.....	359
2.1.18 GDSN Migration Guide for version 3.1.15	360
2.1.18.1 Product 360 data model changes.....	360
2.1.18.2 Units of measure	365
2.1.18.3 Data quality rule and configuration changes	374
2.1.18.4 Changes in data source export templates	381
2.1.18.5 Compatibility Export Template for older versions	400
2.1.18.6 B2B.....	403
2.2 GDSN Accelerator with OpenAS2	404
2.2.1 Why using OpenAS2?	404
2.2.1.1 Having everything in one place	404
2.2.2 Supported Scenarios	404
2.2.3 Official OpenAS2 documentation.....	404
2.2.4 FieldList	405
2.2.5 Choreography.....	405
2.2.5.1 IM message choreography.....	405
2.2.5.2 The choreography "send Catalogue Request item" in more detail.....	406
2.2.5.3 The choreography "receive Catalogue Response item" in more detail	407
2.2.5.4 Other messages.....	407
2.2.6 Installation / Configuration of OpenAS2	408
2.2.6.1 Install OpenAS2.....	408
2.2.6.2 Configure OpenAS2	408
2.2.6.3 Certificates	410
2.2.6.4 Troubleshooting.....	410
2.2.6.5 Certificates	411
2.2.7 Configuration of Product360	414
2.2.7.1 Import Hotfolder configuration	414
2.2.7.2 Export	418

2.2.8	Migration Guide.....	423
2.2.8.1	Export template migration	423
3	Informatica BPM Accelerator.....	424
3.1	Resources	424
3.2	Required Workflows.....	425
3.2.1	Configuration and Installation of BPM.....	426
3.2.2	Informatica BPM Installation.....	426
3.2.2.1	Installation of the Informatica BPM service.....	426
3.2.2.2	Webserver and Java.....	427
3.2.2.3	Integrated Security	428
3.2.3	Installation of the required default workflows.....	429
3.2.4	Configure Dispatch Services.....	432
3.2.5	Preparing Informatica BPM service for JMS based communication	434
3.2.5.1	Download additional library.....	434
3.2.5.2	Setup messaging service within Informatica BPM	435
3.3	Step Workflow	438
3.3.1	Preface.....	438
3.3.2	Prerequisites	439
3.3.3	Installation	439
3.3.4	Configuration	439
3.3.5	StepWorkflow XML files	440
3.3.5.1	The StepWorkflowMap.xml file	441
3.3.5.2	The StepWorkflow.xml file.....	443
3.3.6	Examples	446
3.3.6.1	Steps to run the examples	446
3.3.6.2	Example 1 (1 Task)	448
3.3.6.3	Example 2 (2 Tasks).....	448
3.3.6.4	Example 3 (2 Tasks with DQ checks)	448
3.3.6.5	Example 4 (5 Tasks with DQ checks, parallel and sequential)	448
3.3.6.6	Example 5 (extending example 4 with an approval step)	448
3.3.6.7	Example 6 (2 tasks (incl. 1 supplier task) + modifying of fields)	448
3.3.6.8	Example 7 (2 Tasks + modifying of fields + approval step)	448
3.3.6.9	Example 8 (2 Tasks modifying of fields approval step merge)	448
3.3.6.10	Example 9 (Classic task).....	448

3.3.7	Defaults.....	453
3.3.7.1	DefaultArticle.....	453
3.3.7.2	DefaultVariant	455
3.3.7.3	DefaultProduct2G	457
3.3.8	Example 1 (1 Task)	458
3.3.8.1	Diagram for example 1.....	459
3.3.8.2	Explanation of the steps example 1	459
3.3.8.3	Product 360 prerequisites for example 1.....	460
3.3.8.4	StepWorkflow.xml file for example 1	460
3.3.8.5	Detailed explanation of the steps	461
3.3.9	Example 2 (2 Tasks).....	462
3.3.9.1	Diagram for example 2.....	463
3.3.9.2	Product 360 prerequisites for example 2.....	464
3.3.9.3	Explanation of the steps example 2	464
3.3.9.4	StepWorkflow.xml file for example 2	464
3.3.9.5	Detailed explanation of the steps	465
3.3.10	Example 3 (2 Tasks with DQ checks)	466
3.3.10.1	Diagram for example 3.....	467
3.3.10.2	Product 360 prerequisites for example 3.....	467
3.3.10.3	Explanation of the steps example 3	467
3.3.10.4	StepWorkflow.xml file for example 3	468
3.3.10.5	Detailed explanation of the steps	469
3.3.11	Example 4 (5 Tasks with DQ checks, parallel and sequential)	470
3.3.11.1	Diagram for example 4.....	471
3.3.11.2	Product 360 prerequisites for example 4.....	471
3.3.11.3	Explanation of the steps example 4	472
3.3.11.4	StepWorkflow.xml file for example 4	472
3.3.11.5	Detailed explanation of the steps	474
3.3.12	Example 5 (extending example 4 with an approval step)	477
3.3.12.1	Diagram for example 5.....	478
3.3.12.2	Product 360 prerequisites for example 5.....	478
3.3.12.3	Explanation of the steps example 5	479
3.3.12.4	StepWorkflow.xml file for example 5	479
3.3.12.5	Detailed explanation of the steps	482
3.3.13	Example 6 (2 tasks (incl. 1 supplier task) + modifying of fields)	485

3.3.13.1	Diagram for example 6.....	486
3.3.13.2	Product 360 prerequisites for example 6.....	487
3.3.13.3	Explanation of the steps example 6	487
3.3.13.4	StepWorkflow.xml file for example 6	487
3.3.13.5	Detailed explanation of the steps	488
3.3.14	Example 7 (2 Tasks + modifying of fields + approval step)	492
3.3.14.1	Diagram for example 7.....	493
3.3.14.2	Product 360 prerequisites for example 7	494
3.3.14.3	Explanation of the steps example 7	494
3.3.14.4	StepWorkflow.xml file for example 7	494
3.3.14.5	Detailed explanation of the steps	496
3.3.15	Example 8 (2 Tasks modifying of fields approval step merge)	499
3.3.15.1	Diagram for example 8.....	500
3.3.15.2	Product 360 prerequisites for example 8.....	501
3.3.15.3	Explanation of the steps example 8	501
3.3.15.4	StepWorkflow.xml file for example 8	501
3.3.15.5	Detailed explanation of the steps	504
3.4	Creating MQ based workflows.....	504
3.4.1	REST versus JMS based workflows	505
3.4.2	Mandatory Permission Required For Queue Based Communication.....	506
3.4.3	Types of interactions and their representation in the WSDL file	506
3.4.4	Router process	508
3.4.5	Core orchestration project	508
3.4.6	Correlation	509
3.4.7	Defining participants	509
3.4.7.1	Consumers.....	509
3.4.7.2	Producers	509
3.4.8	Workflow interaction examples	511
3.4.8.1	Workflow instance creation.....	511
3.4.8.2	P360 Service API oneway calls.....	512
3.4.8.3	P360 Service API calls with result handling	512
3.4.8.4	P360 DQ execution with result handling.....	514
3.4.8.5	P360 workflow tasks	514
4	User Interface Templates	515
4.1	Content.....	515

4.2	Import	515
4.3	Approval UI	515
4.4	Text Mastering UI	516
4.5	Media Assignment UI	517
4.6	Classification UI	518
4.7	Attribute Mastering UI	519
5	Web Search Examples.....	520
5.1	Items-only.ext	520
5.2	Products-only.ext.....	520
5.3	Variant.only.ext	520
5.4	All Supplier Catalogs.ext.....	521
5.5	2PPD.ext	521
5.6	3PPD.ext	521
5.7	Steps to use it:	521
6	CLAIRE Accelerator.....	522
6.1	Installation	522
6.1.1	Python usage on Windows	522
6.1.1.1	Python download and installation instructions.....	522
6.1.1.2	Python Machine Learning libraries installation.....	526
6.1.1.3	Folder creation	526
6.1.1.4	CLAIRE server startup	526
6.1.2	Python usage on Linux.....	527
6.1.2.1	Folder creation	527
6.1.2.2	CLAIRE server startup	527
6.1.3	Claire server Health Check.....	527
6.2	Configuration and Operation	528
6.2.1	Configuration of CLAIRE recommendation service	528
6.2.1.1	Preferences.....	529
6.2.2	AI training	530
6.2.2.1	Supported content languages for model generation	530
6.2.2.2	Standard export templates for AI training.....	531
6.2.2.3	Create own export templates for AI training	531
6.2.2.4	Create AI Training export profiles	531

6.2.2.5	Start AI Training	532
6.2.2.6	Display available models	534
6.2.2.7	Natural language processing and deep learning explained	534
6.2.3	Batch classification	534
6.2.3.1	How to use the Auto classification rule configuration	535
6.2.4	Configuration of CLAIRE UI	537
6.3	Best Practices and Recommendations	539
6.3.1	Folder structure for model maintenance	539
6.3.2	Recommendations for best results	539
6.3.3	Model accuracy measurement	540
7	Swagger UI.....	541
7.1	Purpose.....	541
7.2	Installation	541
7.2.1	Prerequisites	541
7.2.2	Terms.....	542
7.2.3	Download package	542
7.2.4	Operation as a Windows Service	542
7.2.5	Operation in Standalone Mode	543
7.3	Configuration and Operation	544
7.3.1	Configuration	544
7.3.2	Operation	545
7.3.2.1	Example : Fetch 'Article' details at Entity Level, by Items in a catalog.....	548
7.4	Best Practices, Recommendations and Known Issues	551
8	Micrometer Metrics and Dashboard.....	551
8.1	Installation, Configuration and Operation	551
8.1.1	Installation	551
8.1.1.1	Accelerator package	551
8.1.2	Configuration	552
8.1.2.1	ElasticSearch Configuration	552
8.1.2.2	Metrics Configuration	553
8.1.2.3	Import of Kibana Saved Objects.....	555
8.2	Dashboards	555
8.2.1	Global Filters	556
8.2.1.1	Application Stack	556

8.2.1.2	Application Server.....	557
8.2.1.3	Entity Type.....	557
8.2.1.4	Entity.....	557
8.2.1.5	Database.....	558
8.2.1.6	Message Type	558
8.2.2	Application Metrics Dashboard	558
8.2.2.1	Job Framework Metrics	558
8.2.2.2	Data Quality Metrics.....	559
8.2.2.3	Service API Metrics	560
8.2.2.4	Persistence Metrics	560
8.2.2.5	Database Tables Statistics.....	564
8.2.2.6	Message Queue Statistics	565
8.2.2.7	Audit trail Metrics	565
8.2.2.8	Trigger Service Metrics.....	566
8.2.2.9	Import Metrics	567
8.2.3	System Metrics Dashboard	568
8.2.3.1	Host Metrics.....	568
8.2.3.2	JVM Statistics - Memory	568
8.2.3.3	JVM Garbage Collection Statistics.....	569
8.2.3.4	Login Session Statistics	570
8.2.3.5	Jetty (Web Service Framework) Statistics	571
8.2.3.6	Connection Pool Statistics	571
8.2.3.7	Thread Pool Statistics.....	573
8.2.4	System Metrics Dashboard - Extended	576
8.2.4.1	Cache Statistics	576
8.2.4.2	Hibernate Statistics.....	578
8.2.4.3	Log4j2 Logging Statistics	580
8.3	Best Practices and Recommendations	581

Accelerators are functioning implementations which can be used out of the box under specific circumstances only. Usage of these accelerators in the context of the concrete customer's use cases and requirements with the customer's data and within its specific environment might need adaptations like configuration, translation, additional scripting, or programming of add-on functionalities which have to be provided through additional professional services.

For example the eCommerce integration provide the fully functioning transfer of a set of data based on the Product 360's standard repository and according to the selected sample implementation of the WebShop. As soon the data set needs to be adapted to the concrete needs of the customer's web-shop or adapting to the customer's individual Product 360 data repository layout additional customization might be needed.

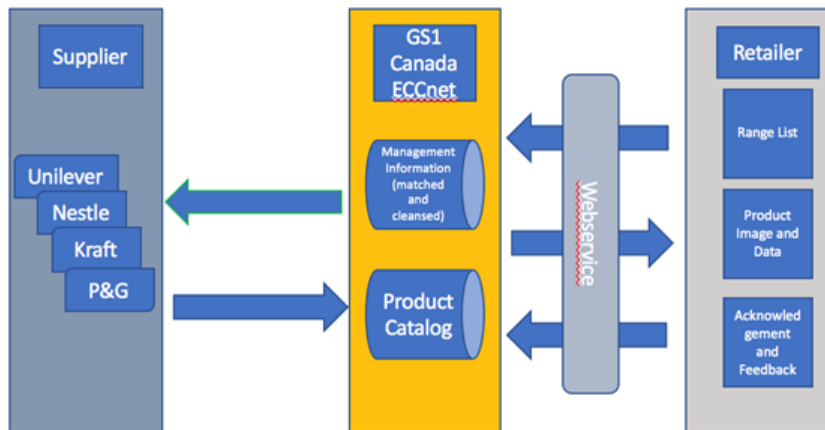
1 ECCnet Accelerator

1.1 Introduction

Throughout the lifecycle of a product, information is needed across many different parties. Manufacturers and brands constantly create and update product information. Retailers need to ensure they have always access to reliable, complete and up-to-date product information to comply with changing regulations and increasing consumer expectations requesting instant access to information through various channels. But product data integrity and quality is at risk due to the silo's nature of applications where this data resides. Retailers selling products are constantly challenged with insufficient, inaccurate product data across their warehouses, stores or online.

GS1 Canada's ECCnet Registry is Canada's national product registry. This product information catalogue service, is the most comprehensive registry of continually validated product data of its kind in Canada. ECCnet is a single point of access between trading partners for electronically sharing accurate product data based on global standards in Canada. By automating the new product listing and product change processes, ECCnet Registry reduces human error, improves product visibility in the supply chain, and speeds access to critical product information between trading partners.

The Accelerator for GS1 Canada's ECCnet allows receiving new or updated product information from suppliers via ECCnet within the MDM – Product 360 business user interface.



1.2 Communicating with ECCnet

Product 360 leverages ECCnet API calls to login, search, subscribe and update product data from ECCnet directly. Only the attributes that are mapped to the defined data model are retrieved to the selected item or items. P360 sends back the appropriate acknowledgement based on whether the item has been subscribed or not.

There are many benefits of using the ECCnet Accelerator:

- Access ECCnet within Product 360 to receive new or updated product information from your suppliers
- Increase efficiency for product listings thanks to automation of product data exchange
- Increase time to market thanks to streamlined collaboration and communication with your suppliers (one-time product data loading only)
- Build brand trust based on GS1 standards
- Increase customer loyalty thanks to accurate, complete and up-to-date product information

1.2.1 Features of the Accelerator

- Built in connectivity to ECCnet registry
- API's to configure additional functionality if required
- Search for items of a manufacturer and retrieve those from the ECCnet data pool
- Schedule daily/weekly/monthly checks for updates on an a catalog and return the list of items that can be updated
- Batch update all ECCnet catalogs or pick an individual catalog
- Turn on privileges for certain suppliers to access, view and execute ECCnet functionality
- Define internally who can access, view and execute ECCnet functionality
- Receive status Messages on success or failure of the calls




1.3 ECCnet Accelerator Features

The ECCnet accelerator consists of three main features. The first one is the data model extension. For ECCnet, we updated our data model and UI's to support the most commonly used attributes by our customers. The second one is a comprehensive list of functionalities to support searching, extracting and

updating of item data from ECCnet registry. The third being support for suppliers to access ECCnet functionality if the customer chooses to. The following sections cover details on these features in detail.

1.4 ECCnet Accelerator package content

The ECCnet accelerator package comes in the folder PIM_<Version>_Rev-xxxxx_ECCnet in a zip file called PIM_<Version>_Rev-xxxxx_Accelerators.zip and contains the following:

-  PIM_8.0.6.01.00_Rev-52576_client_eccnet.delta.zip
-  PIM_8.0.6.01.00_Rev-52576_resources_eccnet.delta.zip
-  PIM_8.0.6.01.00_Rev-52576_server_eccnet.delta.zip

While the client and server zip contain the ECCnet bundles which contains the functionality the "resources" package contains additional content such as repository deltas to adjust the data model, import mappings for structure systems and additional files needed by the accelerator.

1.5 ECCnet Accelerator installation

1.5.1 Prerequisites

1.5.1.1 Product 360 environment

The version of the installed Product 360 must be 8.1.1.04 or higher.

1.5.2 Installing the ECCnet Accelerator

1.5.2.1 Download ECCnet Accelerator zip

The **ECCnet Accelerator** is available via Informatica Shipping. It consists of the folder **PIM_<Version>_ECCNET** in the file **PIM_<Version>_Accelerators.zip**.

1.5.2.2 Extract the ECCnet Accelerator zip


Unpack the file **PIM_<Version>_Accelerators.zip** to a temporary directory on the Product 360 environment. It contains the following artefacts:

- **PIM_<Version>_Rev-xxxxx_client_eccnet.delta.zip**
 - contains PIM Desktop feature and plugins
- **PIM_<Version>_Rev-xxxxx_resources_eccnet.delta.zip**
 - contains import mappings
- **PIM_<Version>_Rev-xxxxx_server_eccnet.delta.zip**
 - contains PIM Desktop feature and plugins

1.5.2.3 Product 360 environment configuration

Product 360 Server

1. Unpack the archive **PIM_<Version>_Rev-xxxxx_server_eccnet.delta.zip** into the Product 360 server folder or use the Product 360 - Control Center.
2. Open the `server.properties` and specify the following ECCnet specific settings:

Setting	Description
<code>eccnet.service.endpoint</code>	URL to ECCnet Item Centre webservice endpoint (mandatory!)
<code>eccnet.autologin.enabled</code>	Specifies if the user is logged in automatically to ECCnet Item Centre or not. If <i>false</i> , the user has to log in manually in the ECCnet search perspective. (default is <i>false</i>)
<code>eccnet.service.user</code>	The user id used for the log in to ECCnet Item Centre (when autologin is enabled, mandatory!)
<code>eccnet.service.password</code>	<p>The password used for the log in to ECCnet Item Centre (when autologin is enabled, mandatory!)</p> <div>  If you want to encrypt the password please refer to chapter Encryption of secure information in the Server Installation manual. </div>
<code>eccnet.service.companyId</code>	The company ID, under which users are registered in ECCnet Item Centre (mandatory!)
<code>eccnet.service.partnerUserId</code>	The partner user ID, for the ECCnet Item Centre user account (optional)

3. Open the `application_modules.properties` file to activate the ECCnet module, see chapter [Application modules for ECCnet](#)(see page 17) for more details.

Repository configuration

In the resources package (PIM_<Version>_Rev-xxxxx_resources_eccnet.delta.zip) in the folder `DataRecipient\configuration\RepositoryDeltas` there are Repository delta files for the Repository.repository

and EnumFragment.repository.

Use the Repository Merger included in the RepositoryEditor in order to apply them to your standard files.

Additionally there are patches for the repository.properties files. You can apply them using for example eclipse (Team → Apply patch ... in the context menu of the file).

Product 360 Desktop Client

Install Product 360 Desktop Client

1. Unpack the archive **PIM_<Version>_Rev-xxxxx_client_eccnet.delta.zip** into the Product 360 client folder.
2. Start the client.

Import GPC Structure

1. Download the latest files from the GS1 website: <http://www.gs1.org/gpc/production>
 - a. It is recommended to download the **Combined Published Schema, or the subset of the required industry**
 - b. The excel file **GS1 Combined Published_Schema as at 0000000.xlsx** from the zip archive is needed
2. Create a new **output** structure system with identifier "GPC". Configure that multiple assignments are not possible for this structure system.
3. Use the import mapping **GPC.en.him** provided in the folder **Common/GPC structure import** of the **PIM_<Version>_Rev-xxxxx_client_resources.delta.zip** to import the downloaded excel
4. Switch the structure system to type "classification system"

Import ECCNET Structure

1. Download the latest version of the **ECCnet Classification Codes.xlsx** file from the website: <http://www.eccnet.profilesys.ca/mdc/GetExcelClassCode/eccc/en>
2. Create a new **output** structure system with identifier "ECCNET". Configure that multiple assignments are not possible for this structure system.
3. Use the import mapping **ECCnet.him** provided in the folder **Common/ECCnet structure import** of the **PIM_<Version>_Rev-xxxxx_resources_eccnet.delta.zip** to import the downloaded excel file
4. Switch the structure system to type "classification system"

1.5.3 Application modules for ECCnet

A module can be seen as business functionality which can be enabled or disabled. The property file `application_modules.properties` located in the configuration folder of the server allows to enable / disable different GDSN and ECCnet scenarios and the food and beverage module. When activated, the corresponding perspectives, views and other things like reports and export datatypes are available for the user to import/maintain/export food and beverage, GDSN or ECCnet data.

By default "GDSN", "ECCnet" as well as "FoodAndBeverage" are set to "false" and must be activated during the accelerator installation (food and beverage only if needed).



The food and beverage module can be used independently of GDSN or ECCnet.

 If ECCnet is used GDSN settings must be set to false.

An example for the `application_modules.properties` file is given in the screenshot below.

```
# Defines if the GDSN extension is installed (true) or not (false)
GDSN = false

# Defines if the GDSN pool "IM" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_im = false


# Defines if the GDSN pool "DSE" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_dse = false

# Defines if the GDSN extension is used in "data source" mode (true) or not (false) if installed.
gdsn_data_source = false

# Defines if the GDSN extension is used in "data recipient" mode (true) or not (false) if installed.
gdsn_data_recipient = false

# Defines if the food and beverage module is activated (true) or deactivated (false).
FoodAndBeverage = true

# Defines if the ECCnet is installed (true) or not (false)
ECCnet = true
```

 Please also note that if ECCnet should be used, of course the corresponding ECCnet plugins need to be installed in the client and server.

1.5.4 Configure ECCnet access from Product 360 - Supplier Portal

It is possible for suppliers and brokers to access ECCnet from Product 360 - Supplier Portal in order to maintain ECCnet items of their catalogs. Thereto, Product 360 - Supplier Portal needs to be configured respectively. Further on a Product 360 - Supplier Portal ECCnet system user needs to be created in Product 360 with respective permissions. This will be explained in the following sections.

1.5.4.1 Configure Product 360 - Supplier Portal

In order to enable ECCnet access from Product 360 - Supplier Portal the following settings in the `configuration.properties` file need to be adjusted:

Setting	
<code>global.permission.ECCnetAccess</code>	Global switch to enable/disable the ECCnet access from Supplier Portal. Possible values:

Setting	
	<ul style="list-style-type: none"> • ENABLED: ECCnet access is enabled. This means that by default ECCnet access is enabled for newly created suppliers. However, the portal administrator can revoke ECCnet access from the supplier. • RESTRICTED: ECCnet access is available, but by default ECCnet access is disabled for newly created suppliers. However, the portal administrator can give ECCnet access to the supplier. • DISABLED: ECCnet access is disabled for the whole Supplier portal. <p>Default: DISABLED</p>
hpm.embedded.itemeditor	<p>Configures not only Item Editor/Item Viewer but also ECCnet access UI to be opened within Supplier Portal or in a separate browser tab.</p> <p>Possible values:</p> <ul style="list-style-type: none"> • true: UI to search in ECCnet will be opened within the same browser tab in which the ECCnet button has been clicked. • false: UI to search in ECCnet will be opened in a separate browser tab. <p>Default: false</p>

1.5.4.2 Configure Product 360 - Supplier Portal ECCnet System User

Product 360 - Supplier Portal uses the Product 360 Web functionality to access ECCnet. Thereto, Product 360 - Supplier Portal requires the credentials of a user belonging to a user group with respective permissions.

Create user group for Product 360 - Supplier Portal ECCnet System User

1. In the *Organization* perspective of the Product 360 Desktop client, create a new user group.
2. Grant at least the following **action rights** to the user group:

Rights group	Permission
Web Permissions	Log in (Product 360 - Web)
ECCnet	ECCnet, general access
Catalogs	Supplier catalogs, general access
Item	Items, general access
Item search	Item search management, general access

Rights group	Permission
GDSN Data	Canada-specific GDSN data, general access

3. **Revoke** the following rights for this group:

Rights group	Permission
Web Permissions	Classify objects (Product 360 - Web)
Web Permissions	Help (Product 360 - Web)
Web Permissions	Change Password
Tasks	Create tasks
Multimedia attachments	Add multimedia attachments



All other action rights not mentioned above, as well as all field rights have to be defined individually depending on the scenario and requirements of the project and the individual use case scenarios

4. Grant at least the following **interface visibility rights** to the user group:

Type	Permission
Web Context	Context visibility: Entire Context selection area
Web Context	Context visibility: ECCnet (Web Access)

5. **Revoke** at least the following interface visibility rights:

Type	Permission
Web Tab	Tab visibility: Product, Variant, References (Product 360 - Web)
Web List Definition	List Definition visibility: of Structuregroup, Product, Variants: Default, Full, Short




All other interface visibility rights not mentioned above, as well as all field rights have to be defined individually depending on the scenario and requirements of the project and the individual use case scenarios

Create Product 360 - Supplier Portal ECCnet System User

1. In the *User* perspective of the Product 360 Desktop client, create a new user.
2. Assign this user to the user group created in the previous step.
3. Make sure that the user is active (*Active* checkbox must be checked for the user in the *User* view).
4. Set the user's *Authentication mode* to *Internal*.

1.5.4.3 Configure Product 360 - Supplier Portal ECCnet System User within Product 360 - Web

The credentials of the Product 360 - Supplier Portal ECCnet System User created in the previous step have to be specified in the `webfrontend.properties` file of the Product 360 server, now. Thereto, configure the following properties:

Setting	
<code>web.client.hsx.eccnet.login</code>	<p>User name of the Product 360 - Supplier Portal ECCnet System User</p> <p>e.g. <code>web.client.hsx.eccnet.login=eccnet</code></p>
<code>web.client.hsx.eccnet.password</code>	<p>Password of the Product 360 - Supplier Portal ECCnet System User</p> <div>  If you want to encrypt the password please refer to chapter Encryption of secure information in the Server Installation manual. </div>

1.6 ECCnet Accelerator operation

The following chapters provide information about the ECCnet operation of:

- [Product 360 - Desktop Client](#)(see page 21)
- [Product 360 - Web Client](#)(see page 36)
- [Product 360 - Supplier Portal](#)(see page 46)

1.6.1 Desktop client

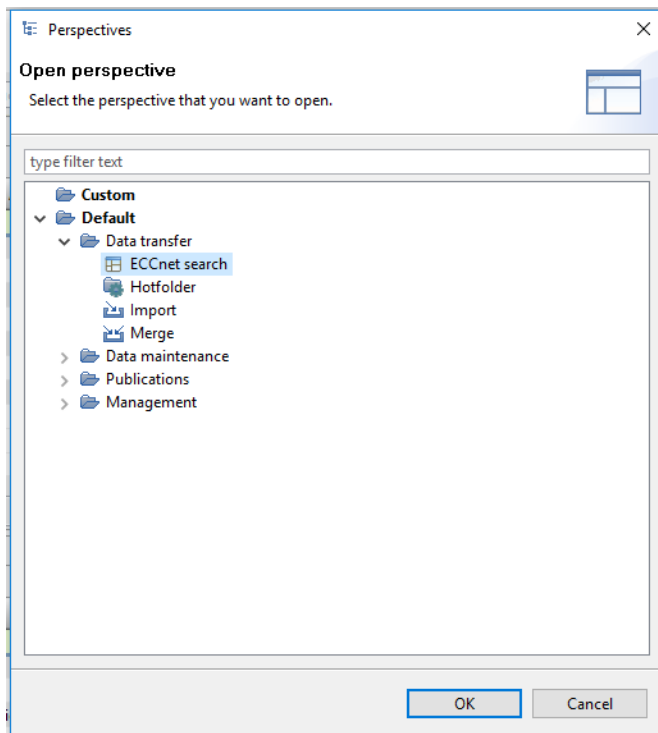
All Product 360 Desktop client users who have access to ECCnet will be able to perform the following operations in the Product 360 Desktop client:

- [Access ECCnet search perspective](#)(see page 22)
- [Search for items in ECCnet Item Centre](#)(see page 23)
- [Browse through the search results](#)(see page 24)

- View packaging hierarchy for items as defined in ECCnet Item Centre(see page 24)
- Subscribe selected items(see page 25)
- Subscribe all the items in the search results from ECCnet Item Centre(see page 27)
- Retrieve (or update) latest data from ECCnet Item Centre for subscribed items(see page 28)
- Unsubscribe items(see page 29)
- View and edit ECCnet items data in Product 360 supplier catalogs(see page 30)
 - ECCnet (core)(see page 31)
 - ECCnet (supply chain data)(see page 31)
- View subscribed items under ECCnet and GPC classification systems(see page 32)
- View packaging hierarchy for subscribed items in Product 360(see page 32)
- Schedule job to update data for all ECCnet items in a Product 360 catalog(see page 33)
 - One-off update(see page 34)
 - Repeated update(see page 35)
- Schedule job to update data for all ECCnet items in all Product 360 catalogs(see page 35)
- View job status and process logs for all recent subscribe/unsubscribe/update data requests (one-off and scheduled)(see page 35)

1.6.1.1 Access ECCnet search perspective

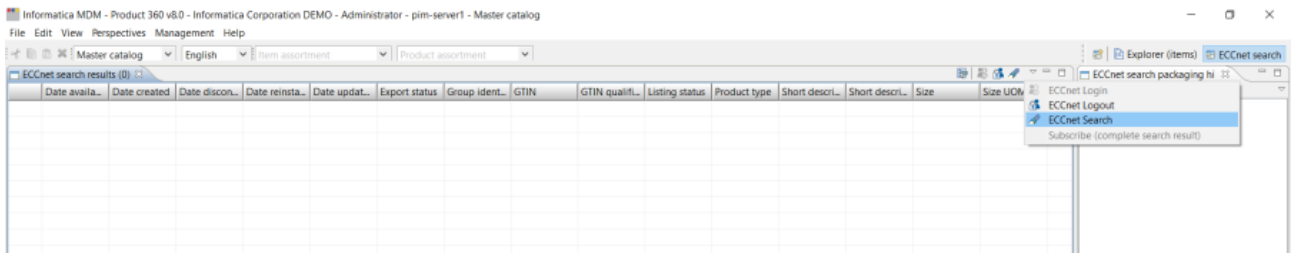
When the *ECCnet, general access* permission is granted to a Product 360 Desktop client user, the user should be able to open the *ECCnet search* perspective located under the *Data transfer* category in the list of perspectives.



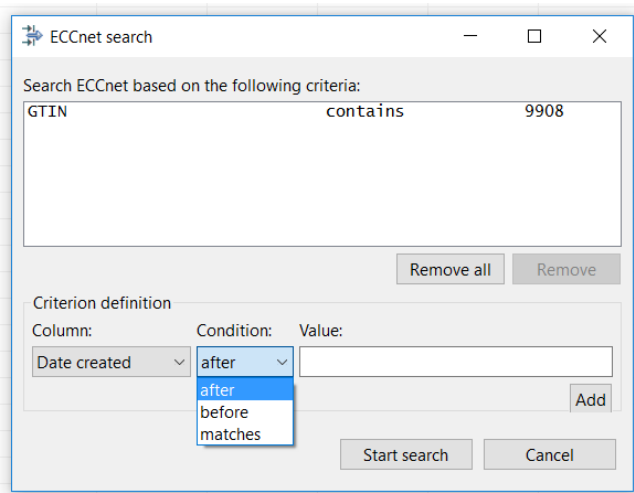
1.6.1.2 Search for items in ECCnet Item Centre

In the *ECCnet search* perspective, users can perform searches for items in ECCnet Item Centre. The searches are based on search queries.

To create a search query, select the *ECCnet search* action from the toolbar or the view menu.



The *ECCnet search* dialog opens allowing to define the search query. A search query consists of one or more search criteria joined by a logical AND. A search criterion consists of a search field a condition operator and a value. E.g. if you want to search for all items with a GTIN containing the number sequence '9908', select 'GTIN' from the *Column* drop-down, select 'contains' from the *Condition* drop-down and enter '9908' into the *Value* field of the *ECCnet search* dialog.



After you have defined a search criterion, click on the *Add* button to add the criterion to the list of search criteria. When you have defined all search criteria, click on the *Start search* button to submit the query to ECCnet Item Centre.

Use the *Remove all* button to clear the list of criteria and create a new criteria list. This does not empty or clear the current search results listing.

Use the *Remove* button to remove any specific criteria from the list. Select the criteria and click on the *Remove* button.

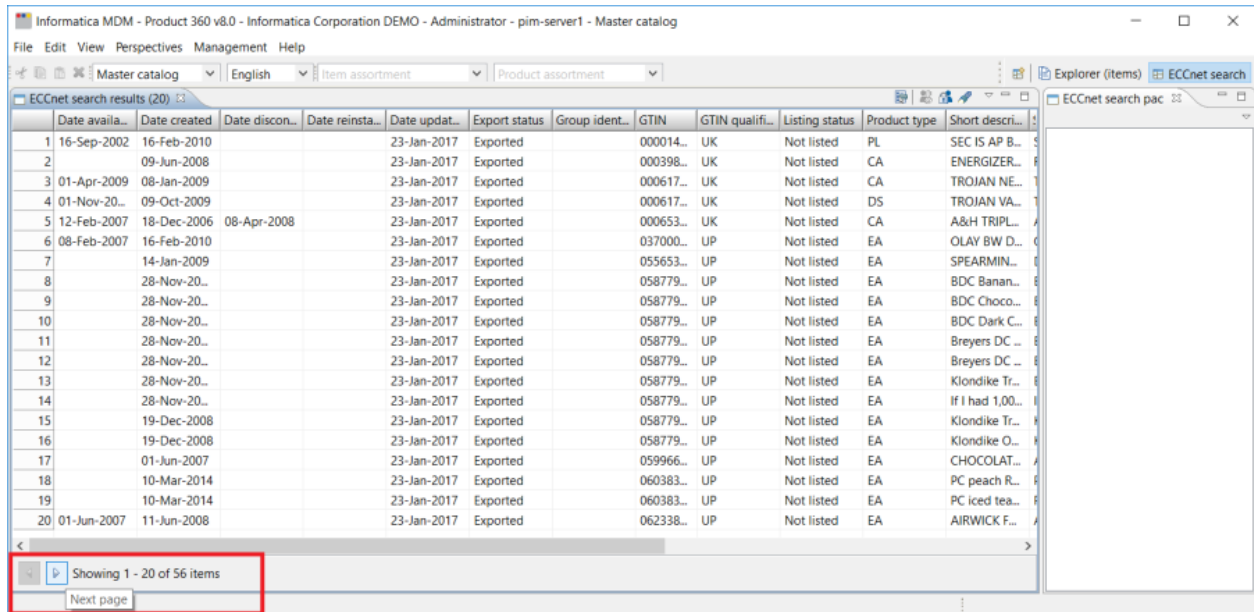
! ECCnet only supports the condition operator 'contains' for non-date fields. Hence, the *Condition* drop-down is disabled for non-date fields with the default value set to "contains".

1.6.1.3 Browse through the search results

ECCnet Item Centre API returns 20 items, at a time displayed in the *ECCnet search results* view.

You can browse through the complete search results by using the *Next/Previous* page buttons at the bottom of view.

The *Next* page button will be disabled if no further search results are available for the search query.



	Date availa...	Date created	Date discon...	Date reinsta...	Date updat...	Export status	Group ident...	GTIN	GTIN quali...	Listing status	Product type	Short descri...
1	16-Sep-2002	16-Feb-2010			23-Jan-2017	Exported		000014...	UK	Not listed	PL	SEC IS AP B...
2		09-Jun-2008			23-Jan-2017	Exported		000398...	UK	Not listed	CA	ENERGIZER...
3	01-Apr-2009	08-Jan-2009			23-Jan-2017	Exported		000617...	UK	Not listed	CA	TROJAN NE...
4	01-Nov-20...	09-Oct-2009			23-Jan-2017	Exported		000617...	UK	Not listed	DS	TROJAN VA...
5	12-Feb-2007	18-Dec-2006	08-Apr-2008		23-Jan-2017	Exported		000653...	UK	Not listed	CA	A&H TRIPL...
6	08-Feb-2007	16-Feb-2010			23-Jan-2017	Exported		037000...	UP	Not listed	EA	OLAY BW D...
7		14-Jan-2009			23-Jan-2017	Exported		055653...	UP	Not listed	EA	SPEARMIN...
8		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not listed	EA	BDC Banan...
9		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not listed	EA	BDC Choco...
10		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not listed	EA	BDC Dark C...
11		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not listed	EA	Breyers DC ...
12		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not listed	EA	Breyers DC ...
13		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not listed	EA	Klondike Tr...
14		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not listed	EA	If I had 1,00...
15		19-Dec-2008			23-Jan-2017	Exported		058779...	UP	Not listed	EA	Klondike Tr...
16		19-Dec-2008			23-Jan-2017	Exported		058779...	UP	Not listed	EA	Klondike O...
17		01-Jun-2007			23-Jan-2017	Exported		059966...	UP	Not listed	EA	CHOCOLAT...
18		10-Mar-2014			23-Jan-2017	Exported		060383...	UP	Not listed	EA	PC peach R...
19		10-Mar-2014			23-Jan-2017	Exported		060383...	UP	Not listed	EA	PC iced tea...
20	01-Jun-2007	11-Jun-2008			23-Jan-2017	Exported		062338...	UP	Not listed	EA	AIRWICK F...

⚠ If the ECCnet session expires due to prolonged inactivity while browsing the search results pages, you will need to perform the search again.

1.6.1.4 View packaging hierarchy for items as defined in ECCnet Item Centre

You can view the complete packaging hierarchy for an item in the search results, regardless of the level the item belongs to.

Select an item in the *ECCnet search results* view to view its packaging hierarchy in the *ECCnet search packaging hierarchy* view.

When more than one items are selected, the *ECCnet search packaging hierarchy* view will be empty.

The screenshot shows the Informatica MDM - Product 360 v8.0 interface. The main window displays 'ECCnet search results (20)' with a table of search results. The table has columns: Date availa..., Date created, Date discon..., Date reinsta..., Date updat..., Export status, Group ident..., GTIN, GTIN qualifi..., and Listing. The results show various items with their respective dates and statuses. On the right side, there is a pane titled 'ECCnet search packaging hierarc...' which shows a hierarchical tree structure of items. The tree is expanded, showing a root node '00001490836575 (Not subscribed)' with several child nodes, some of which are subscribed and some are not. The interface also includes a menu bar with 'File', 'Edit', 'View', 'Perspectives', 'Management', and 'Help'. The status bar at the bottom indicates 'Showing 1 - 20 of 56 items'.

	Date availa...	Date created	Date discon...	Date reinsta...	Date updat...	Export status	Group ident...	GTIN	GTIN qualifi...	Listing
1	16-Sep-2002	16-Feb-2010			23-Jan-2017	Exported		000014...	UK	Not lis
2		09-Jun-2008			23-Jan-2017	Exported		000398...	UK	Not lis
3	01-Apr-2009	08-Jan-2009			23-Jan-2017	Exported		000617...	UK	Not lis
4	01-Nov-20...	09-Oct-2009			23-Jan-2017	Exported		000617...	UK	Not lis
5	12-Feb-2007	18-Dec-2006	08-Apr-2008		23-Jan-2017	Exported		000653...	UK	Not lis
6	08-Feb-2007	16-Feb-2010			23-Jan-2017	Exported		037000...	UP	Not lis
7		14-Jan-2009			23-Jan-2017	Exported		055653...	UP	Not lis
8		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not lis
9		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not lis
10		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not lis
11		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not lis
12		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not lis
13		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not lis
14		28-Nov-20...			23-Jan-2017	Exported		058779...	UP	Not lis
15		19-Dec-2008			23-Jan-2017	Exported		058779...	UP	Not lis
16		19-Dec-2008			23-Jan-2017	Exported		058779...	UP	Not lis
17		01-Jun-2007			23-Jan-2017	Exported		059966...	UP	Not lis
18		10-Mar-2014			23-Jan-2017	Exported		060383...	UP	Not lis
19		10-Mar-2014			23-Jan-2017	Exported		060383...	UP	Not lis
20	01-Jun-2007	11-Jun-2008			23-Jan-2017	Exported		062338...	UP	Not lis



If you don't use the *ECCnet search packaging hierarchy* view frequently, it is recommended to close the *ECCnet search packaging hierarchy* view. This can potentially improve the experience of navigating through the search results as it reduces the number of remote API calls to ECCnet Item Centre for retrieving packaging hierarchy data on real time basis.

1.6.1.5 Subscribe selected items

You can subscribe items found in the *ECCnet search results* and retrieve the complete item data from ECCnet. Thereto, select the desired items in the *ECCnet search results* view and call the *Subscribe (selected items)* functionality from the context menu.

Informatica MDM - Product 360 v8.0 - Informatica Corporation DEMO - Administrator - pim-server1 - Master catalog

File Edit View Perspectives Management Help

Master catalog English Item assortment Product assortment Explorer (items) ECCnet search

ECCnet search results (20)

	Date availa...	Date created	Date discon...	Date reinsta...	Date updat...	Export status	Group ident...	GTIN	GTIN qualifi...	Listing status	Product type	Short descri...	Short descri...
1		09-Jun-2008			13-Mar-2013	Never		000398...	UK	Not listed	CA	EVEREADY ...	AMPOULE ...
2	28-May-20...	18-Nov-20...			18-Nov-20...	Never		000612...	UK	Not listed	CA	CP SINGLE ...	EC REG WU...
3		18-Jul-2005			13-Feb-2012	Never		000693...	UK	Not listed	CA	15/18 HAM...	15/18 EMB...
4	30-Aug-20...	26-Sep-2010			26-Sep-2010	Never		019014...	UP	Not listed	EA	EUK DG 4L...	EUK CHIEN...
5	27-Jan-2010	26-Sep-2010			26-Sep-2010	Never		037000...	UP	Not listed	EA	FEB REFRS...	FEB DESOD...
6	17-Oct-2005	18-Nov-20...	24-Dec-2005		18-Nov-20...	Never		046447...	UP	Not listed	EA	SOAP ON A...	SAVON SU...
7	15-Oct-2004	18-Nov-20...	24-Dec-2004		18-Nov-20...	Never		046447...	UP	Not listed	EA	SOAP ON A...	SAVON SU...
8	15-Dec-2004	10-Mar-2006			15-Dec-2004	Never		059749...	UP	Not listed	EA	Skittles Lip ...	Skittles Bau...
9		25-Jan-2006			15-Dec-2004	Never		059749...	UP	Not listed	EA	CHOCOLAT...	CAPUCHO...
10	18-Sep-2009	18-Sep-2009			18-Sep-2009	Never		056638...	UP	Not listed	EA	SEC FS TOE...	SEC FS C/PL...
11		16-Apr-2007	30-Jun-2008		31-Aug-20...	Never		057961...	UP	Not listed	CA	24pk Sunry...	24pk Sunry...
12		06-Dec-2007			02-Jan-2014	Never		058779...	UP	Not listed	EA	Breyers Sko...	Breyers Sko...
13		10-Mar-2014			10-Mar-2014	Never		059749...	UP	Not listed	EA	MS New Co...	SM Nouvea...
14		10-Mar-2014			10-Mar-2014	Never		059749...	UP	Not listed	EA	MS Authent...	SM Cola di...
15		10-Mar-2014			10-Mar-2014	Never		059749...	UP	Not listed	EA	MS Authent...	SM Cola Au...
16		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	PODRAVKA...	Podravka A...
17		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	PODRAVKA...	Vegeta Ass...
18		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	PODRAVKA...	PODRAVKA...
19		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	Herbal tea ...	The de d'e...
20		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	Herbal tea ...	The de me...

Showing 1 - 20 of 186 items



The *Subscribe (selected items)* functionality will not be enabled, if the item(s) selected has the *Export Status* 'Exported' in ECCnet Item Centre.

When you subscribe items, Product 360 retrieves the complete item data from ECCnet Item Centre and stores it to a supplier catalog. Therefore, it is required to select the supplier catalog, where the retrieved item data should be added to. Select the supplier catalog from the drop-down and click OK.

Informatica MDM - Product 360 v8.0 - Informatica Corporation DEMO - Administrator - pim-server1 - Master catalog

File Edit View Perspectives Management Help

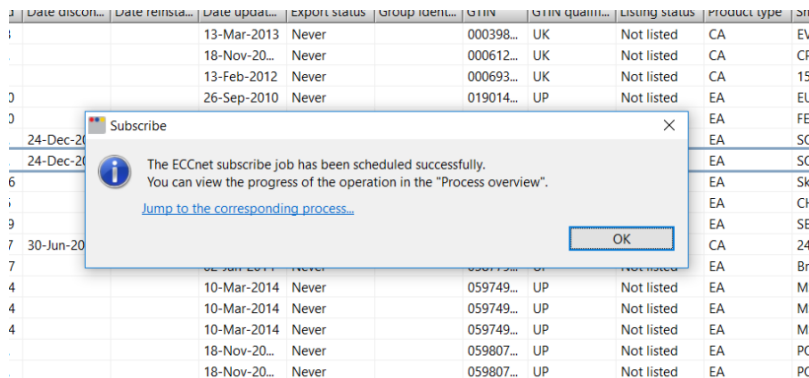
Master catalog English Item assortment Product assortment Explorer (items) ECCnet search

ECCnet search results (20)

	Date availa...	Date created	Date discon...	Date reinsta...	Date updat...	Export status	Group ident...	GTIN	GTIN qualifi...	Listing status	Product type	Short descri...	Short descri...
1		09-Jun-2008			13-Mar-2013	Never		000398...	UK	Not listed	CA	EVEREADY ...	AMPOULE ...
2	28-May-20...	18-Nov-20...			18-Nov-20...	Never		000612...	UK	Not listed	CA	CP SINGLE ...	EC REG WU...
3		18-Jul-2005			13-Feb-2012	Never		000693...	UK	Not listed	CA	15/18 HAM...	15/18 EMB...
4	30-Aug-20...	26-Sep-2010			26-Sep-2010	Never		019014...	UP	Not listed	EA	EUK DG 4L...	EUK CHIEN...
5	27-Jan-2010	26-Sep-2010			26-Sep-2010	Never		037000...	UP	Not listed	EA	FEB REFRS...	FEB DESOD...
6	17-Oct-2005	18-Nov-20...	24-Dec-2005		18-Nov-20...	Never		046447...	UP	Not listed	EA	SOAP ON A...	SAVON SU...
7	15-Oct-2004	18-Nov-20...	24-Dec-2004		18-Nov-20...	Never		046447...	UP	Not listed	EA	SOAP ON A...	SAVON SU...
8	15-Dec-2004	10-Mar-2006			15-Dec-2004	Never		059749...	UP	Not listed	EA	Skittles Lip ...	Skittles Bau...
9		25-Jan-2006			15-Dec-2004	Never		059749...	UP	Not listed	EA	CHOCOLAT...	CAPUCHO...
10	18-Sep-2009	18-Sep-2009			18-Sep-2009	Never		056638...	UP	Not listed	EA	SEC FS TOE...	SEC FS C/PL...
11		16-Apr-2007	30-Jun-2008		31-Aug-20...	Never		057961...	UP	Not listed	CA	24pk Sunry...	24pk Sunry...
12		06-Dec-2007			02-Jan-2014	Never		058779...	UP	Not listed	EA	Breyers Sko...	Breyers Sko...
13		10-Mar-2014			10-Mar-2014	Never		059749...	UP	Not listed	EA	MS New Co...	SM Nouvea...
14		10-Mar-2014			10-Mar-2014	Never		059749...	UP	Not listed	EA	MS Authent...	SM Cola di...
15		10-Mar-2014			10-Mar-2014	Never		059749...	UP	Not listed	EA	MS Authent...	SM Cola Au...
16		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	PODRAVKA...	Podravka A...
17		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	PODRAVKA...	Vegeta Ass...
18		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	PODRAVKA...	PODRAVKA...
19		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	Herbal tea ...	The de d'e...
20		18-Nov-20...			18-Nov-20...	Never		059807...	UP	Not listed	EA	Herbal tea ...	The de me...

Showing 1 - 20 of 186 items

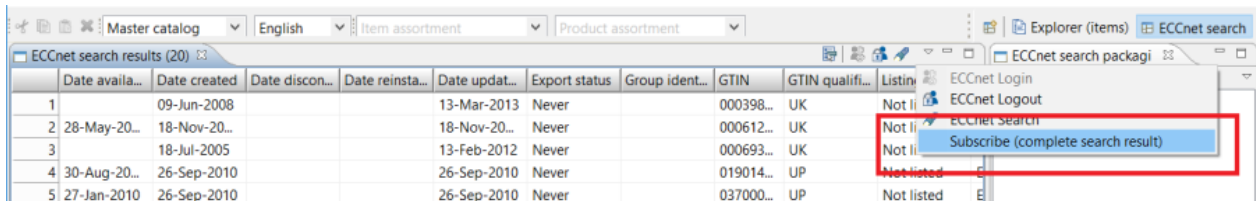
Product 360 will create a server side job to subscribe and retrieve the data for the selected items from ECCnet Item Centre. The following message will be displayed:



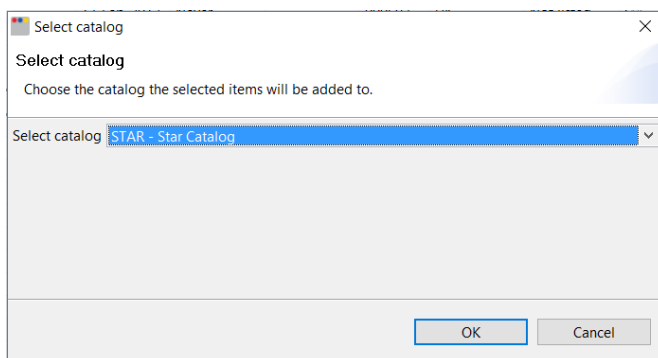
Click on the *Jump to the corresponding process* link to view the progress of the job and related logs (including information messages and error messages) in the *Process Overview* perspective. Once the items are subscribed successfully, they can be looked up within the supplier catalog selected before.

1.6.1.6 Subscribe all the items in the search results from ECCnet Item Centre

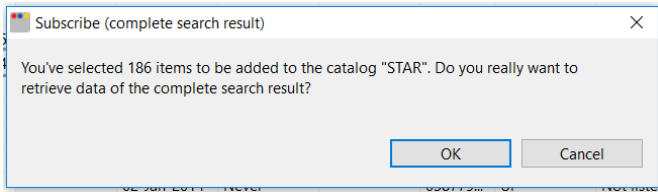
You can also subscribe all the items found in the *ECCnet search results*. Thereto, call the *Subscribe (complete search result)* functionality from the view menu or the context menu.



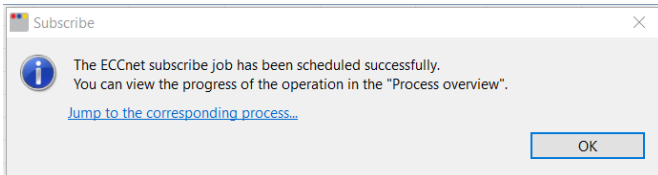
Also here, it is required to select the supplier catalog, where the data of the subscribed items should be added to. Select the supplier catalog from the drop-down and click *OK*.



Confirm by clicking *OK*.



Product 360 will create a server side job to subscribe and retrieve the data for the selected items from ECCnet Item Centre. The following message will be displayed:



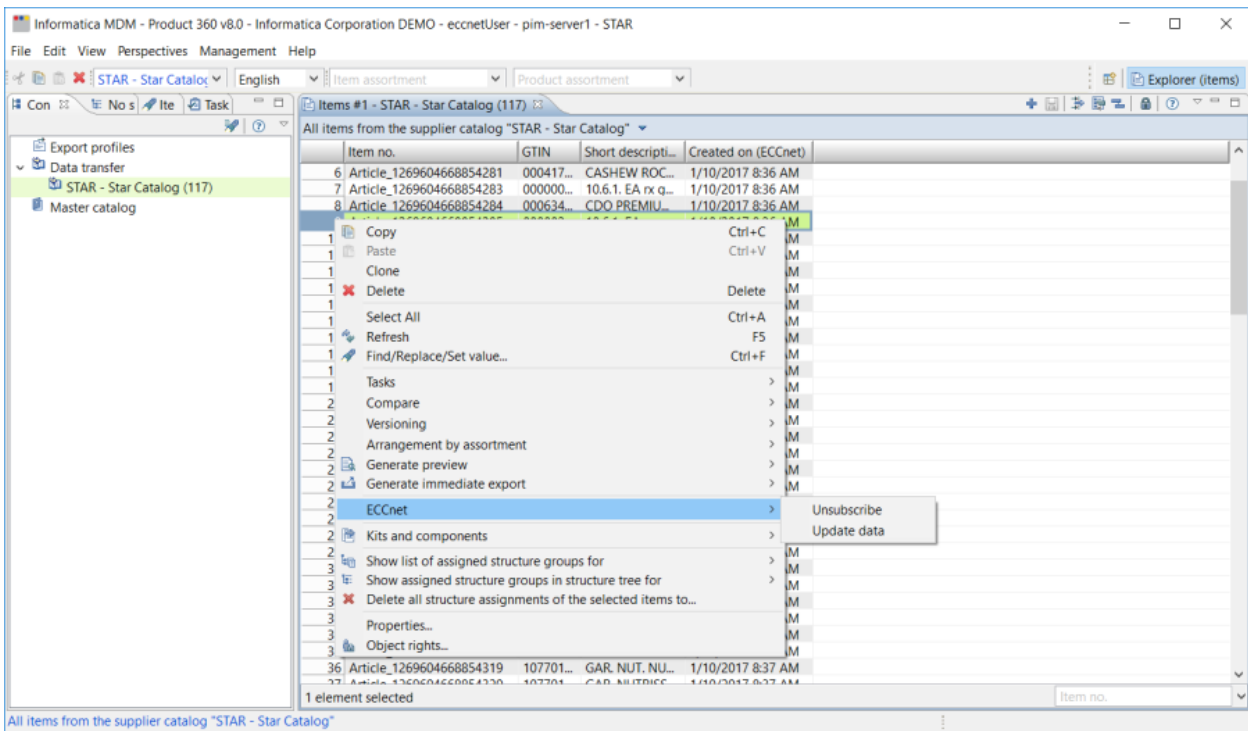
Click on the *Jump to the corresponding process* link to view the progress of the job and related logs (including information messages and error messages) in the *Process Overview* perspective. Once the items are subscribed successfully, they can be looked up within the supplier catalog selected before.

Things to remember when subscribing items in ECCnet Item Centre

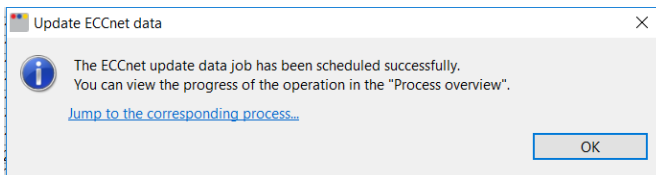
- The *Export status* for the item will be marked as 'Exported' in ECCnet Item Centre, once it is subscribed and the item data are transferred into a Product 360 supplier catalog.
- Items with an *Export status* set to 'Exported' in ECCnet Item Centre cannot be subscribed a second time. Such items can however be updated in the Product 360 supplier catalog. The functionality to update an already subscribed item is explained in following sections.
- A GTIN can be loaded by multiple suppliers in ECCnet Item Centre. The only restriction is that a GTIN can only exist once in each supplier catalog of Product 360.
- Several field level validations and constraints can be set up in the Product 360 repository. The item is not created in Product 360 supplier catalog, in case any of the fields' data from ECCnet Item Centre fails due to such constraints defined in the Product 360 repository. All these errors are tracked in the problem logs and can be viewed in the *Process Overview* perspective in the Product 360 Desktop client.
- Product 360 does not allow to add the item data to the master catalog at subscription.

1.6.1.7 Retrieve (or update) latest data from ECCnet Item Centre for subscribed items

ECCnet Item Centre does not update automatically items which have been subscribed by Product 360 and have been updated in ECCnet Item Centre in the meantime. However, you can retrieve the updated data by means of the *Update data* functionality. This functionality is available in the context menu of views listing items of supplier catalogs, e.g. the *Items#1* view of the *Explorer (items)* perspective.



Select the items you want to update and choose *ECCnet > Update data* from the context menu. Product 360 creates a server job and the following message is displayed.



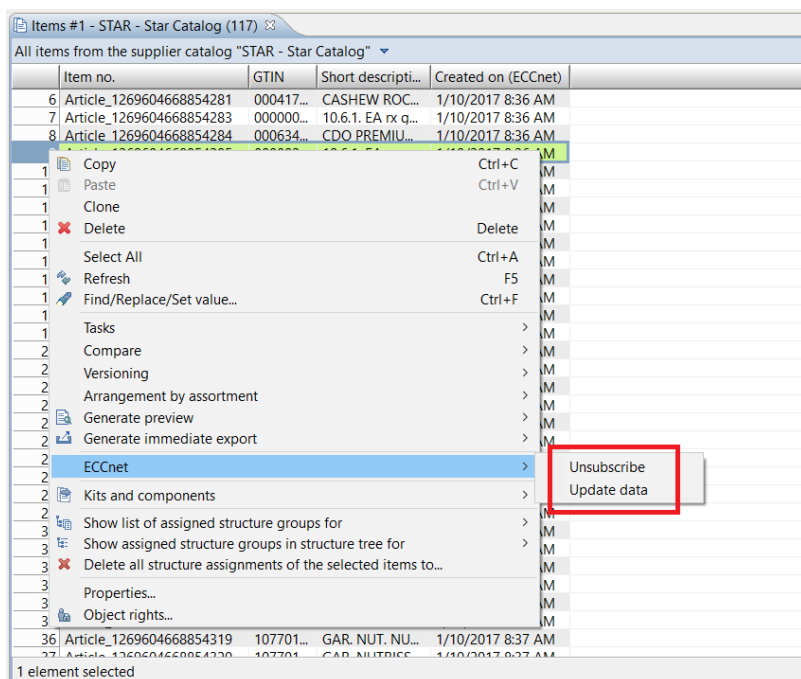
Click on the *Jump to the corresponding process* link in the message to view the progress of the job and related logs (including information messages and error messages) in the *Process Overview* perspective.



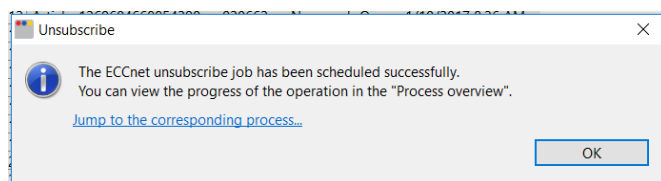
Instead of updating subscribed items manually, you can also schedule an update job updating all subscribed items of a supplier catalog or even of all supplier catalogs. This will be explained below.

1.6.1.8 Unsubscribe items

You can also unsubscribe already subscribed items in ECCnet Item Centre, to stop receiving any updates of the item in Product 360. The *Unsubscribe* functionality is available in the context menu of views listing items of supplier catalogs, e.g. the *Item#1* view of the *Explorer (items)* perspective.



Select the items you want to unsubscribe and choose *ECCnet > Unsubscribe* from the context menu. Product 360 creates a server job and the following message will be displayed.



Click on the [Jump to the corresponding process](#) link in the message to view the progress of the job and related logs (including information messages and error messages) in the *Process Overview* perspective.

1.6.1.9 View and edit ECCnet items data in Product 360 supplier catalogs

There are two perspectives available to view and edit ECCnet specific fields in the Product 360 Desktop client.

ECCnet (core)

Informatica MDM - Product 360 v8.0 - Informatica Corporation DEMO - eccnetUser - pim-server1 - STAR

File Edit View Perspectives Management Help

STAR - Star Catalog English Item assortment Product assortment Explorer (items) ECCnet (core)

Items #1 - STAR - Star Catalog (117)

All items from the supplier catalog "STAR - Star Catalog"

Item no.	GTIN	Short descripti...	Created on (ECCnet)
102	Article_284480564941424	507750...	1/23/2017 3:33 PM
103	Article_284480564941425	717489...	1/23/2017 3:33 PM
104	Article_284480564941426	717489...	1/23/2017 3:33 PM
105	Article_284480564941427	717489...	1/23/2017 3:33 PM
106	Article_284480564941428	717489...	1/23/2017 3:33 PM
107	Article_284480564941429	717489...	1/23/2017 3:33 PM
108	Article_284480564941430	717489...	1/23/2017 3:33 PM
109	Article_284480564941431	775026...	1/23/2017 3:33 PM
110	Article_861976375758018	046447... SOAP ON A R...	1/25/2017 10:42 A...
111	Article_861976375758021	059807... Fettuccine eqq...	1/25/2017 10:57 A...
112	Article_861976375758019	100598... GRANDE FIDE...	1/25/2017 10:57 A...
113	Article_861976375758020	100598... FETTUCCINE E...	1/25/2017 10:57 A...
114	Article_861976375758024	100598... Pickled red pe...	1/25/2017 10:57 A...
115	Article_861976375758023	059807... Pickled red pe...	1/25/2017 10:57 A...
116	Article_861976375758022	100598... Pickled ovpsv...	1/25/2017 10:57 A...
117	Article_861976375758025	059807... GRANDE FIDE...	1/25/2017 10:57 A...

1 element selected

Target market specific ECCnet data Item references GS1 trade item identification keys Trade item lifespan ECCnet specific data (1) Promotional item Security tag Sales

Article_861976375758025 - GRANDE FIDELINI EGG NOODLES

Production type	Division	Packaged in Canada	Prepared in Canada	Processed in Canada	Product of Canada	Refined in Canada
1 Manufactured		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

1 element selected

ECCnet (supply chain data)

Informatica MDM - Product 360 v8.0 - Informatica Corporation DEMO - eccnetUser - pim-server1 - STAR

File Edit View Perspectives Management Help

STAR - Star Catalog English Item assortment Product assortment Explorer (items) ECCnet (core) ECCnet (supply chain data) Process overview

Items #1 - STAR - Star Catalog (117)

All items from the supplier catalog "STAR - Star Catalog"

Item no.	GTIN	Short descripti...	Created on (ECCnet)
79	Article_1269604668854326	200653... EXCELLENCE P...	1/10/2017 8:37 AM
80	Article_861976375758021	059807... Fettuccine eqq...	1/25/2017 10:57 A...
81	Article_861976375758020	100598... FETTUCCINE E...	1/25/2017 10:57 A...
82	Article_1269604668854319	107701... GAR. NUT. NU...	1/10/2017 8:37 AM
83	Article_1269604668854338	770103... GAR. NUT. NU...	1/10/2017 8:37 AM
84	Article_1269604668854320	107701... GAR. NUTRISS...	1/10/2017 8:37 AM
85	Article_1269604668854340	770103... GAR. NUTRISS...	1/10/2017 8:37 AM
86	Article_861976375758019	100598... GRANDE FIDE...	1/25/2017 10:57 A...
87	Article_861976375758025	059807... GRANDE FIDE...	1/25/2017 10:57 A...
88	Article_1269604668854306	069618... GRANTHAMS ...	1/10/2017 8:37 AM
89	Article_1269604668854318	100696... GRANTHAMS ...	1/10/2017 8:37 AM

1 element selected

Packaging-spec Packaging mark Trade item han Data carrier info Delivery purchas Handling instruc Packaging mate Additional trad Health wellness Packaging mark

Article_861976375758025 - GRANDE FIDELINI EGG NOODLES

Packaging form	Packaging material

0 elements

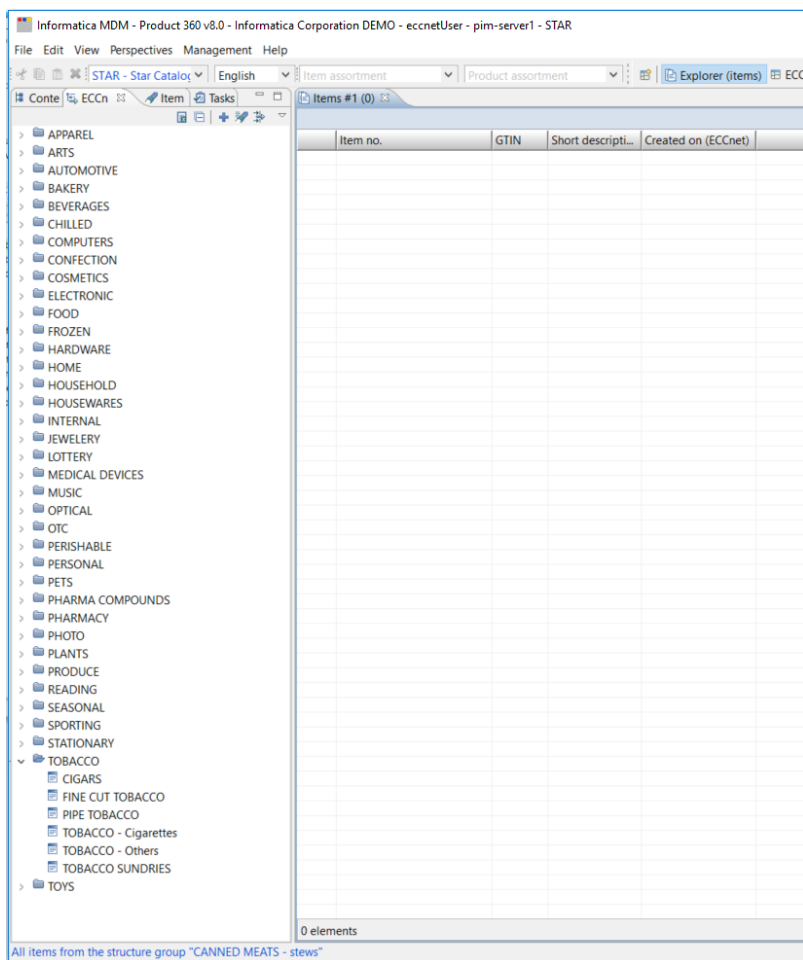
0 elements

1.6.1.10 View subscribed items under ECCnet and GPC classification systems

Product 360 Desktop client users will be able to view subscribed items classified under the GPC and ECCnet classification systems.



Make sure that the GPC and ECCnet classification systems are defined and the appropriate structures are imported into Product 360, before you subscribe any items.



1.6.1.11 View packaging hierarchy for subscribed items in Product 360

Product 360 Desktop client users can view the product packaging hierarchy for subscribed items. The hierarchy in Product 360 will mirror the packaging hierarchy in ECCnet Item Centre, if all the items of the hierarchy are subscribed. To view the product packaging hierarchy in Product 360 Desktop, make sure that the *ECCnet packaging hierarchy* view is visible in the current perspective. Select an item e.g. in the *Items #1* view to view the hierarchy structure in the *ECCnet packaging hierarchy* view.

The screenshot displays the Informatica MDM interface. On the left, the 'Data transfer' menu is highlighted under 'Export profiles'. Below it, the 'ECCnet packaging hierarchy' view shows a reference hierarchy for 'Article_861976375758023'. The main pane shows a table of items from the 'STAR - Star Catalog' (117). The table has columns: Item no., Article, GTIN, Short description (English), and Product type. Item 101, 'Article_861976375758023', is selected and highlighted in blue. Its description is 'Pickled red peppers' and its product type is 'Base unit or each'.

Item no.	Article	GTIN	Short description (English)	Product type
94	Article_1269604668854291	00065338...	LOP EVERSTRONG MCKESSON FLO...	Case
95	Article_1269604668854301	00064893...	LOP SHIPPER OF 44 VOLUMINOUS...	Case
96	Article_1269604668854290	02066200...	Newman's Own Marinara Sauce	Base unit or each
97	Article_1269604668854293	04644704...	NFW COL SPRAY 9 ML	Base unit or each
98	Article_1269604668854272	00059300...	Peter Jackson MEF KSSS25	Case
99	Article_861976375758022	10059807...	Pickled qypsy salad	Case
100	Article_861976375758024	10059807...	Pickled red peppers	Case
101	Article_861976375758023	05980710...	Pickled red peppers	Base unit or each
102	Article_1269604668854309	10026888...	QATestAutomatedTestingONLY- D...	Display shipper
103	Article_1269604668854300	05980039...	QUALITY STREET CLEAR BOX 180G	Base unit or each
104	Article_1269604668854288	00064100...	Raisin Bran Crmch Q3A Promo 50	Case
105	Article_1269604668854339	83312500...	RIZ Rice Shells	Case
106	Article_1269604668854292	03574660...	ROC D Purif-AC Purifying Clean	Case
107	Article_1269604668854299	05800030...	S/SOAP ANTI+RAIN 225ML 2PK	Base unit or each
108	Article_1269604668854329	47760456...	SH D ENG	Base unit or each
109	Article_1269604668854333	44074567...	Short Description ENG	Case
110	Article_1269604668854336	30997149...	SMART SHADE BAGB PPKB	Case
111	Article_1269604668854313	10062000...	Smoked bf Ham Loaf Shingle	Case
112	Article_861976375758018	04644798...	SOAP ON A ROAP	Base unit or each
113	Article_1269604668854302	06713134...	STEAKHOUSE SLICED BLACK OLIV...	Base unit or each
114	Article_1269604668854315	10067131...	STEAKHOUSE SLICED BLACK OLIV...	Case
115	Article_1477035859177940	00025900...	Swisher Kings	Case
116	Article_1269604668854295	00069618...	TW SHIP BLK 3 VARIETY	Display shipper

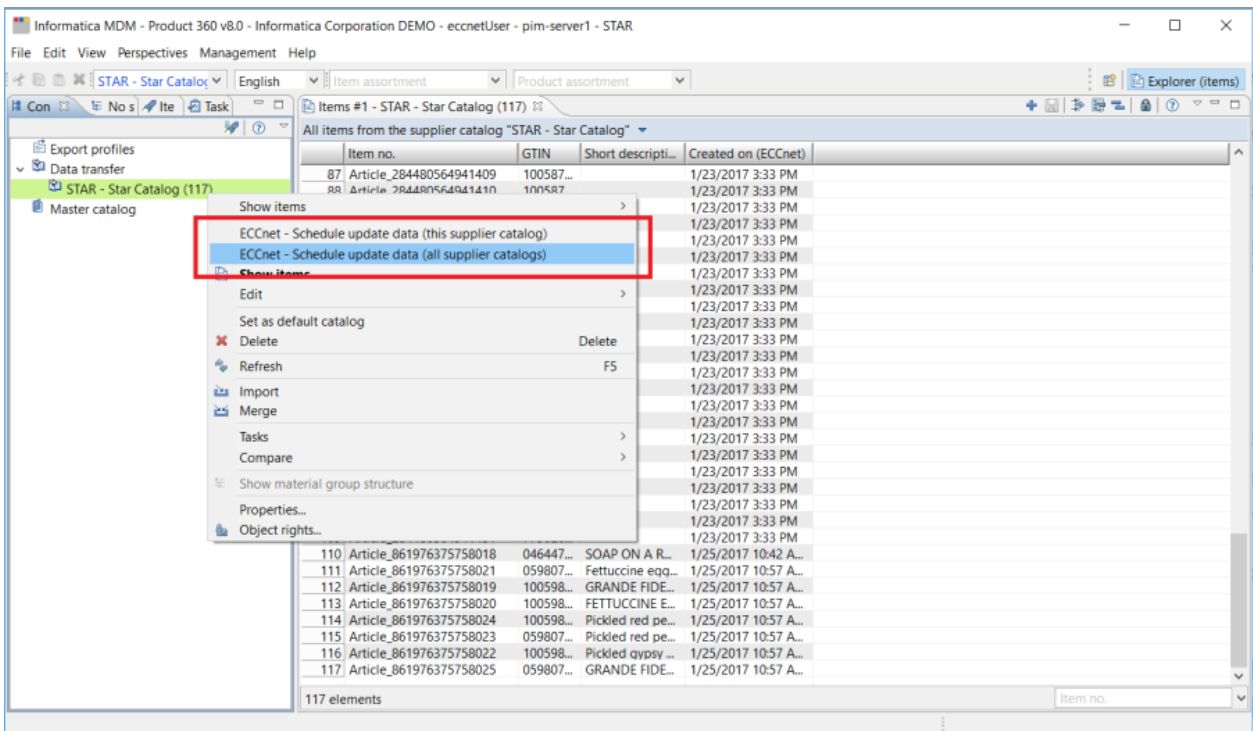
1 element selected



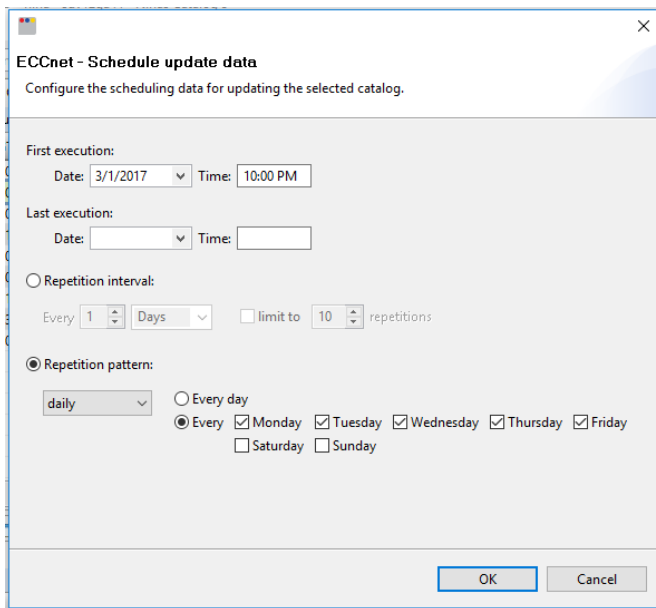
Please note that the difference between the *ECCnet packaging hierarchy* view and the *ECCnet search packaging hierarchy* view is the data source. The *ECCnet packaging hierarchy* view displays items which are present in the Product 360 system (= subscribed items) whereas the source of the *ECCnet search packaging hierarchy* view is the ECCnet data pool (= will include not subscribed items)

1.6.1.12 Schedule job to update data for all ECCnet items in a Product 360 catalog

Instead of updating manually items which have been subscribed by Product 360 and have been updated in ECCnet Item Centre in the meantime, you can also schedule jobs which update these items automatically. You can schedule such jobs by means of the *Schedule update data (this supplier catalog)* functionality.



In the *Context selection* view, select the supplier catalog you want to update and choose *ECCnet > Schedule update data (this supplier catalog)* from the context menu. The following dialog will pop up:



Here you can either schedule an one-off update or a repeated update job.

One-off update

If you want to schedule an one-off update, you will just have to specify the *Date* and the *Time* of the *First execution* section.

Repeated update

If you want to schedule a repeated update job, you can either specify the update times by means of a *Repetition interval* or a *Repetition pattern*.

The *Repetition interval* represents a repeated update at equal intervals and an end date. The first update is performed in accordance with the *Date* and *Time* details in the *First execution* section. Subsequent updates are performed at the exact intervals specified. The number of repetitions can also be specified.

A *Repetition pattern* enables you to make your update times even more flexible than is possible with a *Repetition interval*. You can schedule your updates for particular days of the week, weeks, and months of the year or hours and minutes of the day. The pattern starts from the date of the *First execution*. For example, you want to execute an update daily on working days, but not at the weekend.

1.6.1.13 Schedule job to update data for all ECCnet items in all Product 360 catalogs

You can also schedule a job that updates all ECCnet items in all Product 360 catalogs by means of the *Schedule update data (all supplier catalogs)* functionality. In the *Context selection* view, select any supplier catalog and choose *ECCnet > Schedule update data (all supplier catalogs)* from the context menu and proceed as described above for the update of a single catalog.



Updating all supplier catalogs is a very expensive operation. Therefore, it is recommended to run this at a time where Product 360 is used by a low amount of users, e.g. at night.

1.6.1.14 View job status and process logs for all recent subscribe/unsubscribe/update data requests (one-off and scheduled)

Product 360 Desktop users can view the process logs for all the server jobs including subscribe, unsubscribe, scheduled catalog updates and subscribe search results, in the *Process Overview* perspective.

Num.	User name	Date	Status	Progress	Process info	Information	Execution server
65	142 eccnetUser	1/10/2017 2:03 ...	Completed	100%	Subscribe items	STAR	pim-server1 (Ad...
66	141 eccnetUser	1/10/2017 2:03 ...	Completed	100%	Subscribe items	STAR	pim-server1 (Ad...
67	140 eccnetUser	1/10/2017 2:02 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
68	139 eccnetUser	1/10/2017 2:01 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
69	138 eccnetUser	1/10/2017 1:59 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
70	137 eccnetUser	1/10/2017 1:48 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
71	136 eccnetUser	1/10/2017 1:47 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
72	135 eccnetUser	1/10/2017 1:40 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
73	134 eccnetUser	1/10/2017 1:16 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
74	133 eccnetUser	1/10/2017 1:11 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
75	132 eccnetUser	1/10/2017 1:10 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
76	131 eccnetUser	1/10/2017 1:01 ...	Completed (with errors)	100%	Subscribe items	STAR	pim-server1 (Ad...
77	127 eccnetUser	1/10/2017 12:51 ...	Completed	100%	Subscribe items	STAR	pim-server1 (Ad...
78	125 eccnetUser	1/9/2017 11:54 ...	Completed	100%	Subscribe items	STAR	pim-server1 (Ad...
79	123 eccnetUser	1/9/2017 11:07 ...	Completed	100%	Subscribe items	STAR	pim-server1 (Ad...

Status	Category	Data type	Message	Sequence	Position	Number
1	Summary		ECCnet subscribe job started	0		
2	Summary	Item	1 of 1 objects of data type 'Item' successfully processed	1		1

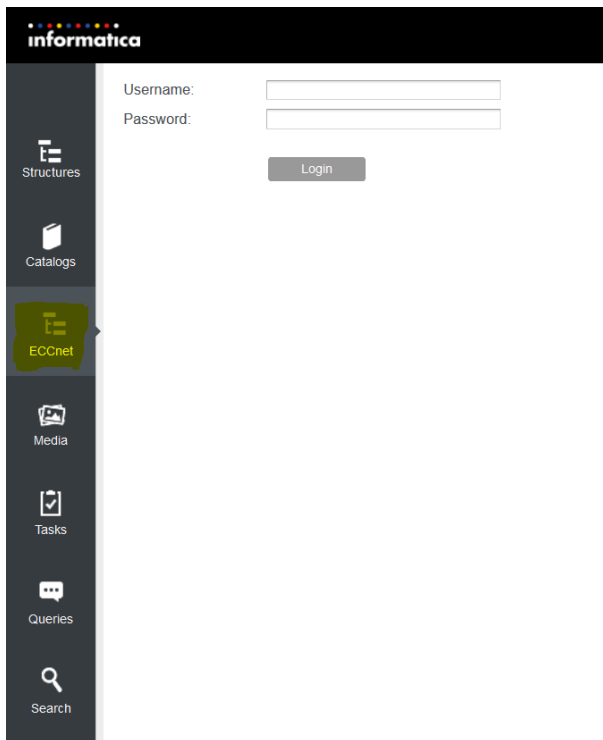
1.6.2 Web client

All Product 360 Web client users who have access to ECCnet will be able to perform the following operations in the Product 360 Web client:

- [Access ECCnet](#)(see page 36)
- [Search for items in ECCnet Item Centre](#)(see page 37)
- [Browse through the search results](#)(see page 38)
- [View packaging hierarchy for items as defined in ECCnet Item Centre](#)(see page 39)
- [Subscribe selected items](#)(see page 39)
- [Subscribe all the items in the search results](#)(see page 40)
- [Retrieve \(or update\) latest data from ECCnet Item Centre for subscribed items](#)(see page 41)
- [Unsubscribe items](#)(see page 42)
- [View subscribed items in Product 360 under ECCnet and GPC classification systems](#)(see page 43)
- [View and edit ECCnet items data in Product 360](#)(see page 44)
- [View packaging hierarchy for subscribed items in Product 360](#)(see page 45)

1.6.2.1 Access ECCnet

When the *ECCnet, general* access permission is granted to a Product 360 Web client user, the user should be able to access ECCnet via the *ECCnet* button in the navigation area. If the ECCnet functionality is not configured in auto-login mode, the user will have to enter his login credentials for ECCnet before he gets to the ECCnet search view.



The screenshot shows the Informatica MDM Product 360 10.1 Accelerators interface. On the left is a dark sidebar with a vertical list of icons and labels: Structures, Catalogs, ECCnet (highlighted with a green background), Media, Tasks, Queries, and Search. The main content area is white and contains a login form. The form has two input fields: 'Username:' and 'Password:'. Below these fields is a 'Login' button. The Informatica logo is visible in the top left corner of the main area.



The ECCnet login credentials must be registered at ECCnet Item Centre under the same company profile that is configured in the *ECCnet Accelerator* section of your Product 360 server configuration.

1.6.2.2 Search for items in ECCnet Item Centre

In the *ECCnet* area, users can perform searches for items in ECCnet Item Centre. The searches are based on search queries. A search query consists of one or more search criteria joined by a logical AND. A search criterion consists of a search field a condition operator and a value.

Search queries and search criteria can be defined in the left area of the ECCnet area. E.g. if you want to search for all items with a GTIN containing the number sequence '9908', create a search criterion in the following way: select 'GTIN' from the drop-down on the left of this area, select 'contains' from the drop-down on the right and enter '9908' into the text field below.

Click on the *Add* button to add the criteria to the list below. Once the list is defined completely, click on the *Search* button to submit the query to ECCnet Item Centre. Use the *Reset* button to clear the list of search criteria and to create a new criteria list.

The screenshot shows the Informatica MDM interface. On the left is a navigation menu with icons for Structures, Catalogs, ECCnet, Media, Tasks, Queries, and Search. The main area is divided into two sections. The top section contains a 'Catalog' dropdown menu, a 'GTIN' dropdown menu, a 'contains' dropdown menu, and an 'Add' button. Below these is a 'Search' button and a 'Reset' button. The bottom section is a table with three columns: 'Field', 'Operator', and 'Value'. The 'Field' column contains 'GTIN', the 'Operator' column contains 'contains', and the 'Value' column is empty. The 'Actions' dropdown menu is visible at the top right of the main area.



ECCnet only supports the condition operator 'contains' for non-date fields. Hence, the drop-down for selecting the condition operator is disabled for non-date fields with the default value set to "contains".

1.6.2.3 Browse through the search results

ECCnet Item Centre API returns 20 items, at a time. You can browse through the complete search results by using the *Next/Previous page* buttons at the top of the search results. The *Next page* button will be disabled if no further search results are available for the search query.

The screenshot shows the Informatica MDM interface with search results. The top navigation bar includes the Informatica logo, user name 'Doe Joe', and links for 'View', 'Help', and 'Log out'. Below the navigation bar is a search criteria panel on the left and a table of search results on the right. The search criteria panel shows 'Catalog' set to 'GTIN' and 'Operator' set to 'contains'. The table of search results has 17 columns: 'Date avail', 'Date cre', 'Date dis', 'Date reir', 'Date upc', 'Export s', 'Group ic', 'GTIN', 'GTIN qu', 'Listing s', 'Product', 'Short de', 'Short de', 'Size', 'Size UOI', 'Total un', 'Vendor i', and 'Vendor i'. The table displays 20 items, with the first item being 'Jul 2, 201' and the last item being 'Apr 15, 2'. The table is paginated, showing 'Showing 1 - 20 of 95 items'.

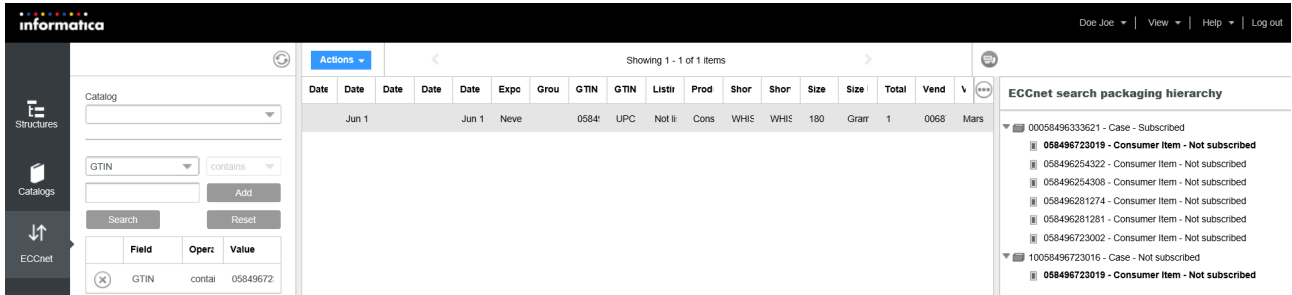
Date avail	Date cre	Date dis	Date reir	Date upc	Export s	Group ic	GTIN	GTIN qu	Listing s	Product	Short de	Short de	Size	Size UOI	Total un	Vendor i	Vendor i
Jul 2, 201	Sep 25, 1			Sep 25, 1	Never		0003700	SCC-14	Not listec	Display s	APDO/P	APDO/P	1	Count	40	0037000	Procter & G
	Nov 18, 1			Nov 18, 1	Never		0006563	SCC-14	Not listec	Mixed Ps	32CT F&	32CT F&	0		234	9052124	General I
	Apr 24, 2			Apr 24, 2	Never		0007017	SCC-14	Not listec	Case	TW MIXE	TW TISA	20	Each	6	0068780	National
	Nov 18, 1			Nov 18, 1	Never		0013344	UPC (GT	Not listec	Consum	3 MUSKI	3 MUSKI	440	Gram	30	0068780	Mars Car
Jun 7, 20	Sep 26, 1			Sep 26, 1	Never		0190140	UPC (GT	Not listec	Consum	IAMS DC	IAMS DC	2.77	Kilogram	1	0037000	Procter & G
	Nov 19, 1			Nov 19, 1	Never		0333444	UPC (GT	Not listec	Consum	PAINT AI	OEUF S /	108	Gram	1	0068780	Mars Car
	Apr 15, 2			Apr 15, 2	Never		0334445	UPC (GT	Not listec	Consum	UB CLAS	UB CLSK	170	Gram	1	0068780	Mars Car



If the ECCnet session expires due to prolonged inactivity while browsing the search results pages, you will need to perform the search again.

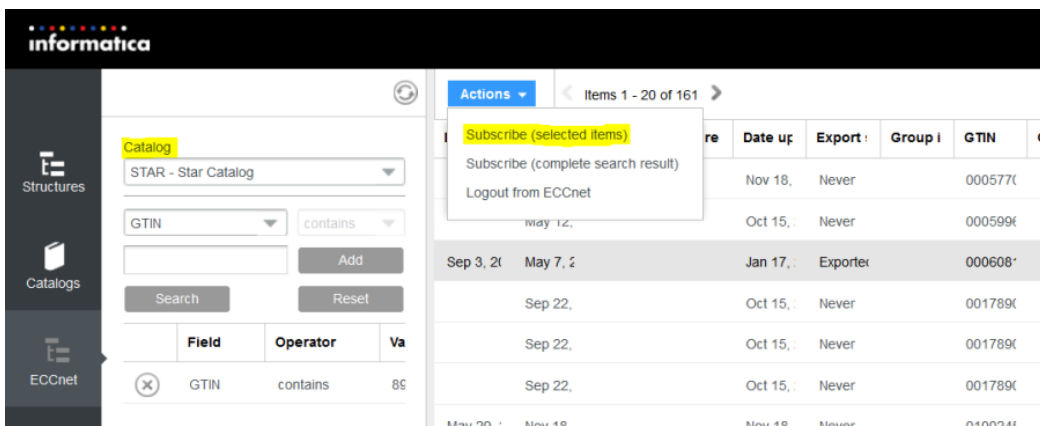
1.6.2.4 View packaging hierarchy for items as defined in ECCnet Item Centre

You can view the complete packaging hierarchy for an item in the search results, regardless of the level the item belongs to. Open the *ECCnet search packaging hierarchy* view via the *ECCnet search packaging hierarchy* button in the upper right corner. Select an item in the table to view its packaging hierarchy in the *ECCnet search packaging hierarchy* view. When more than one items are selected, the *ECCnet search packaging hierarchy* view will be empty.



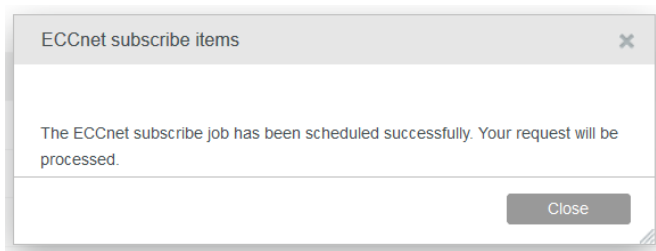
1.6.2.5 Subscribe selected items

You can subscribe items found in the *ECCnet search results* and retrieve the complete item data from ECCnet and store it to a supplier catalog. Therefore, it is required to select the supplier catalog, where the retrieved item data should be added to. Select the supplier catalog from the *Catalog* drop-down in the left area of the ECCnet area. Then select the desired items in the *ECCnet search results* and call the *Subscribe (selected items)* functionality from the *Actions* menu.



The *Subscribe (selected items)* functionality will not be enabled, if the item(s) selected has the *Export Status* 'Exported' in ECCnet Item Centre.

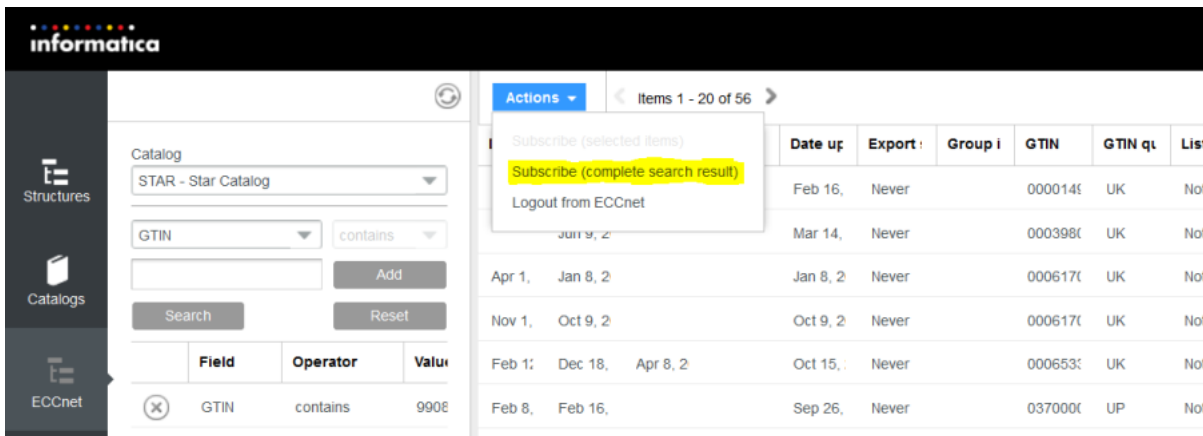
Product 360 will create a server side job to subscribe and retrieve the data for the selected items from ECCnet Item Centre. The following message will be displayed:



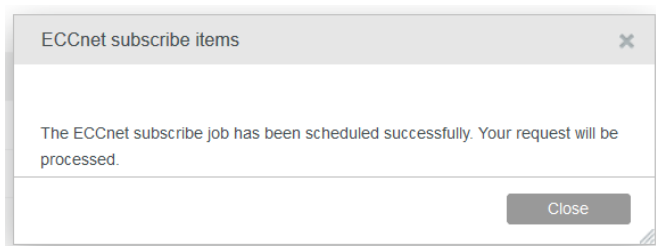
Once the item is subscribed successfully, it can be looked up within P360 supplier catalog. In case of any errors, Product 360 administrators can see the detailed errors for each item in respective problem logs. Presently, Product 360 Web users are not notified about the successful completion or errors from the server.

1.6.2.6 Subscribe all the items in the search results

You can also subscribe all the items found in the *ECCnet search results*. Thereto, call the *Subscribe (complete search result)* functionality from the **Actions** menu.



Product 360 will create a server side job to subscribe and retrieve the data for the selected items from ECCnet Item Centre. The following message will be displayed:



As the server job runs and the items are subscribed successfully, they can be looked up within the Product 360 supplier catalog. In case of any errors, Product 360 administrators can see the detailed errors for each item in the respective problem logs.

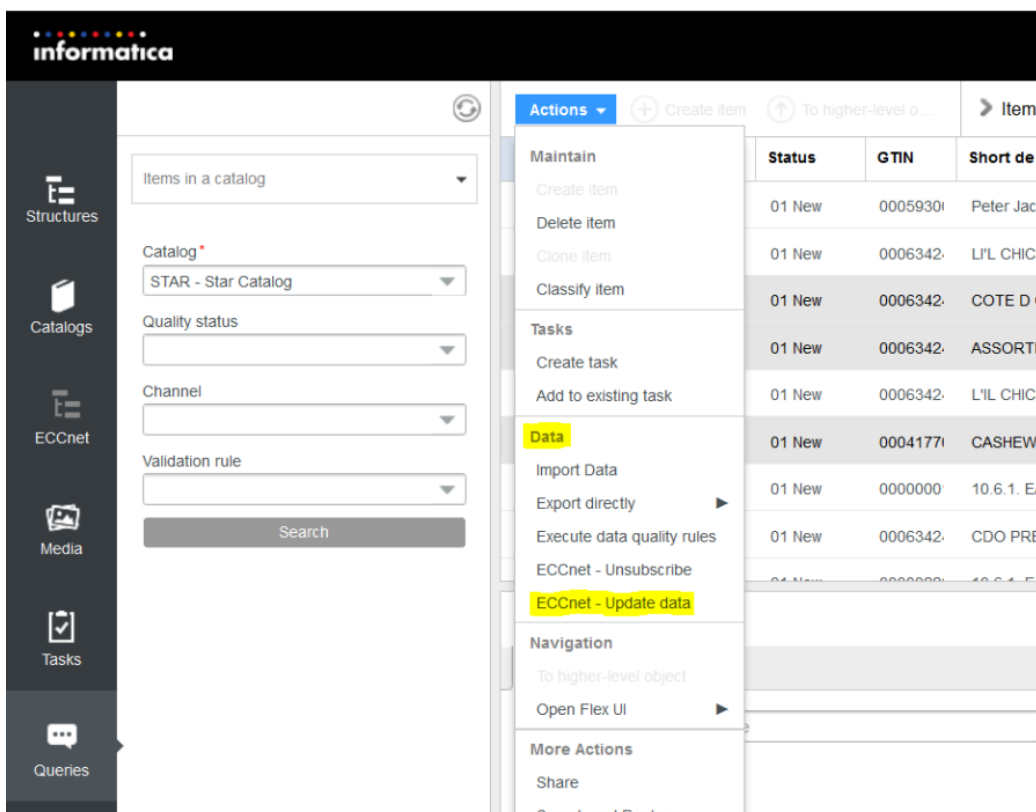


- The *Export status* for the item will be marked as 'Exported' in ECCnet Item Centre, once it is subscribed and the item data are transferred into a Product 360 supplier catalog.

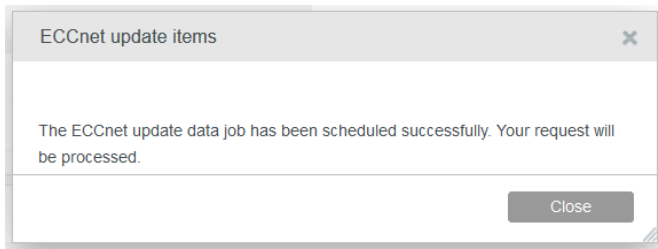
- Items with an *Export status* set to 'Exported' in ECCnet Item Centre cannot be subscribed a second time. Such items can however be updated in the Product 360 supplier catalog. The functionality to update an already subscribed item is explained in following sections.
- A GTIN can be loaded by multiple suppliers in ECCnet Item Centre. The only restriction is that a GTIN can only exist once in each supplier catalog of Product 360.
- Several field level validations and constraints can be set up in the Product 360 repository. The item is not created in Product 360 supplier catalog, in case any of the fields' data from ECCnet Item Centre fails due to such constraints defined in the Product 360 repository. All these errors are tracked in the problem logs and can be viewed in the *Process Overview* perspective in the Product 360 Desktop client.
- Product 360 does not allow to add the item data to the master catalog at subscription.

1.6.2.7 Retrieve (or update) latest data from ECCnet Item Centre for subscribed items

ECCnet Item Centre does not update automatically items which have been subscribed by Product 360 and have been updated in ECCnet Item Centre in the meantime. However, you can retrieve the updated data by means of the *Update data* functionality. This functionality is available in the *Actions* menu of table views listing items of supplier catalogs, e.g. in the *Catalogs* or the *Queries* area.



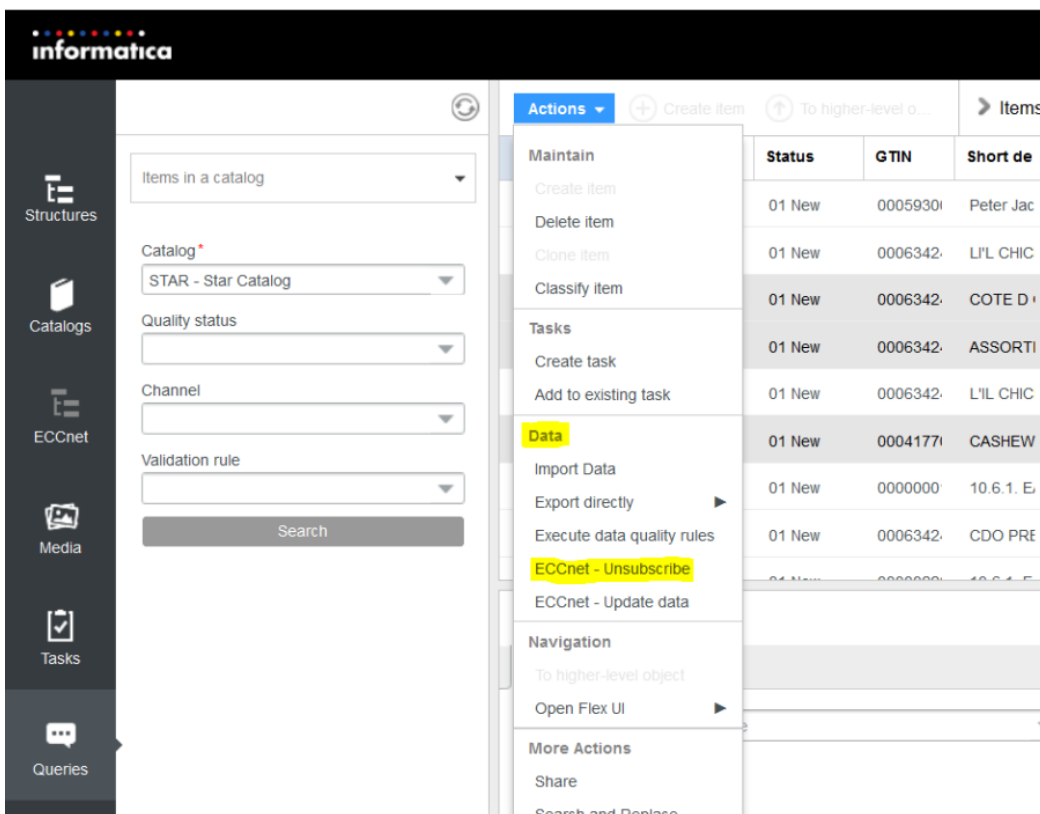
Select the items you want to update and choose *ECCnet – Update data* from the *Actions* menu. Product 360 creates a server job and the following message is displayed:



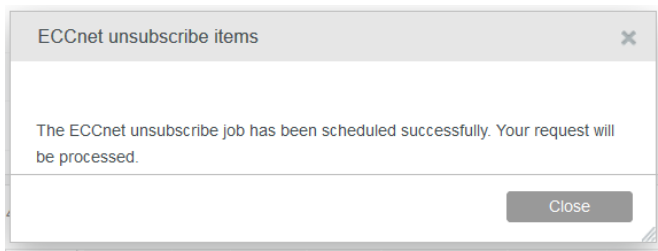
As the server job runs and the items are updated successfully, the updated data can be viewed in the corresponding supplier catalog. In case of any errors, Product 360 administrators can see the detailed errors for each item in respective problem logs. Presently, Product 360 Web client users are not notified about successful completion or errors from the server job.

1.6.2.8 Unsubscribe items

You can also unsubscribe already subscribed items in ECCnet Item Centre, to stop receiving any updates of the item in Product 360. The *Unsubscribe* functionality is available in the *Actions* menu of table views listing items of supplier catalogs, e.g. in the *Catalogs* or the *Queries* area.



Select the items you want to unsubscribe and choose *ECCnet – Unsubscribe* from the *Actions* menu. Product 360 creates a server job and the following message is displayed:



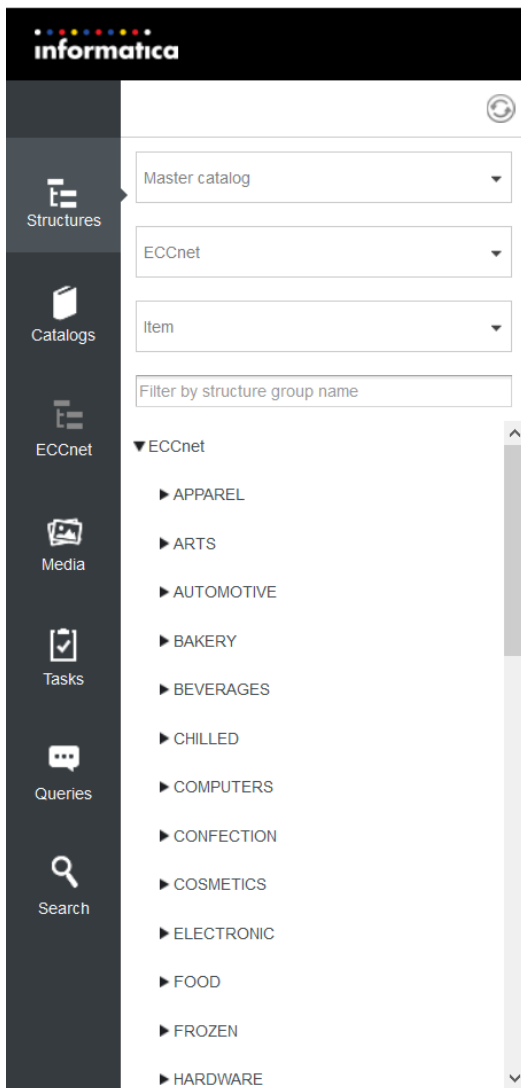
In case of any errors, Product 360 administrators can see the detailed errors for each item in respective problem logs. Presently, Product 360 Web client users are not notified about successful completion or errors from the server job.

1.6.2.9 View subscribed items in Product 360 under ECCnet and GPC classification systems

Product 360 Web users will be able to view subscribed items classified under GPC and ECCnet classification system.



Make sure that the GPC and ECCnet classification systems are defined and the structures imported into Product 360, before any items are subscribed.



1.6.2.10 View and edit ECCnet items data in Product 360

Product 360 Web client users can view and edit ECCnet Item Centre specific fields for subscribed items in several tabs.

Actions

Create item

To higher-level ob...

ECCNet - ECCNet (53)

Item no.

Status


GTIN

Short description (English)

Created on (ECCNet)

Y_item	01 New	056600808676	Y	2/2/2017 2:09 PM
Article_26701037347974	01 New			
Article_26701037347977	01 New	717226616044	MOIST.SHAMP.H-E 2X250ML	2/6/2017 11:14 AM
Article_199339473642256	01 New	00058496333621	TEMPTATIONS 180G HALF PALLET	3/7/2017 3:11 PM
Article_285732256346339	01 New	058496723019	WHISKAS TEMPTATIONS SALMON	3/8/2017 2:41 PM

Item "Article_285732256346339 - WHISKAS TEMPTATIONS SALMON"



Item no.:

Article_285732256346339

Status:

01 New

Nutrient information

Nutritional claims

Preparation serving information

Serving quantity information

ECCnet data (base)

ECCnet data (core)

ECCnet data (extended)

Help

Orderable unit:

No

Height:

20.000

Shipping unit:

No

Height UOM:

Centimeter

Invoice unit:

No

Width:

4.000

Package marked returnable:

No

Width UOM:

Centimeter

Discontinue date:

No content

Length:

16.000

Origin:

No content

Length UOM:

Centimeter

Retail pack:

1

Gross weight:

195.000

Coupon family code:

No content

Gross weight UOM:

Gram

Pre-priced product:

No

Net weight:

180.000

Manufacturer:

Effem Inc.

Net weight UOM:

Gram

Replaces GTIN:

No content

Volume:

0.001

Purchase type:

No content

Volume UOM:

Cubic Meter

1.6.2.11 View packaging hierarchy for subscribed items in Product 360

You can view the product packaging hierarchy for subscribed items. The hierarchy in Product 360 will mirror the packaging hierarchy in ECCnet Item Centre, if all the items of the hierarchy are subscribed. To view the product packaging hierarchy in Product 360 Desktop, click the *ECCnet Packaging Hierarchy* button in the upper right corner to open the *ECCnet Packaging Hierarchy* view. Select an item in the table to view the hierarchy structure in the *ECCnet packaging hierarchy* view.

informatica

Doe Jbe | View | Help | Log out

Master catalog

Data transfer

Filter by catalog name

AAA - Lieferantenkatalog für Test

dc1

ECCnet_PIMEXT-17

ECCnet_PIMEXT-67

ECCNet - ECCNet

ECCNetTest

Actions

Create item

To higher-level ob...

ECCNet - ECCNet (53)

Item no.

Status

GTIN

Shor

Created on (ECCNet)

deleted_test_3	01 New	987456321784	Cook	1/25/2017 4:22 PM
SHRIMP	01 New	00013800445551	LC S	1/26/2017 3:06 PM
deleted_test_1	01 New	11111111111120	Crea	1/30/2017 11:37 AM
Article_199362215830505	01 New	00013800419200	STFF	
Y_item	01 New	056600808676	Y	2/2/2017 2:09 PM
Article_26701037347974	01 New			
Article_26701037347977	01 New	717226616044	MOIS	2/6/2017 11:14 AM
Article_199339473642256	01 New	00058496333621	TEMI	3/7/2017 3:11 PM
Article_285732256346339	01 New	058496723019	WHIS	3/8/2017 2:41 PM

ECCnet packaging hierarchy

Actions

00058496333621 - Case - TEMPTATIONS 180G HALF PALLET

058496723019 - Consumer item - WHISKAS TEMPTATIONS SALMON



Please note that the difference between the *ECCnet packaging hierarchy* view and the *ECCnet search packaging hierarchy* view is the data source. The *ECCnet packaging hierarchy* view displays items which are present in the Product 360 system (= subscribed items) whereas the source of the *ECCnet search packaging hierarchy* view is the ECCnet data pool (= will include not subscribed items)

1.6.3 Supplier Portal

It is possible for suppliers and brokers to access ECCnet from Product 360 - Supplier Portal in order to maintain ECCnet items of their catalogs.



In order to use this functionality, Product 360 - Supplier Portal has to be configured respectively.

1.6.3.1 Configure ECCnet access for a supplier

If you are a portal administrator, you will be able to grant ECCnet access to suppliers and brokers acting for these suppliers. The ECCnet access is configured in the *Supplier Details* view. Click the *Edit* button in the *Configuration for ECCnet access* section.

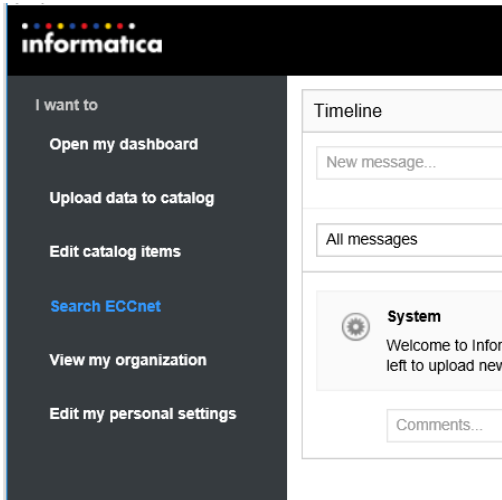
The screenshot shows the 'Supplier Details' view in the Informatica Supplier Portal. The left sidebar contains navigation links: 'I want to', 'Open my dashboard', 'Manage imports', 'View suppliers' (highlighted), 'Invite suppliers', 'View mappings', and 'Edit my personal settings'. The main content area is titled 'Supplier Details' and includes a 'To supplier overview' button. The 'Supplier name' is 'ECCNet' and the 'Supplier identifier' is 'ECCNet'. The 'Configuration for automatic import' shows 'Manual approval required' with an 'Edit' link. The 'Configuration for online data editing' shows 'Online editing of catalog data enabled' with an 'Edit' link. The 'Configuration for ECCnet access' section is highlighted in yellow and shows 'Accessing ECCnet disabled' with an 'Edit' link. Below this is a 'List of Users' table with columns: Name, E-mail, State, User role, and Action. The table contains one user: 'ECCNET QA2 ECC Net User' with email 'qa-pim-sp-11@mailtest.local', state 'Active', and user role 'Broker'. The 'Action' column has a 'Choose action' dropdown.

Select the *Accessing ECCnet enabled* option in the emerging dialog and Save your changes.

The screenshot shows the 'Supplier Details' view in the Informatica Supplier Portal, similar to the previous one. A dialog box titled 'Configuration for ECCnet access for ECCNet' is open in the foreground. The dialog has two radio buttons: 'Accessing ECCnet enabled' (which is selected) and 'Accessing ECCnet disabled'. At the bottom of the dialog are 'Save' and 'Close' buttons. The background shows the same 'Supplier Details' view with the 'Configuration for ECCnet access' section highlighted in yellow.

1.6.3.2 Access ECCnet

When ECCnet access is granted to a supplier, the supplier's users should be able to access ECCnet via the *Search ECCnet* button in the navigation area.



If the ECCnet functionality is not configured in auto-login mode, the user will have to enter his login credentials for ECCnet before he gets to the ECCnet search view. In the ECCnet search view he can:

- Search for items in ECCnet Item Centre and
- Subscribe items and add them to his supplier catalogs (see chapter [ECCnet Accelerator operation > Web client](#) (see page 36))

1.7 ECCnet Accelerator field list

A complete list of the ECCnet fields provided by the ECCnet Accelerator can be found in the EXCEL file **ECCNet_FieldList.xlsx**. The EXCEL file is contained in the folder **DataRecipient** of the **PIM_<Version>_Rev-xxxxx_resources_eccnet.delta.zip** shipped in the ECCnet accelerator package **PIM_<Version>_Rev-xxxxx_ECCnet**.

1.8 ECCnet Accelerator limitations

When customizing the ECCnet accelerator please consider the following limitations.

1.8.1 Service API

Read requests with unqualified fields of "ArticleIngredientLang" and "ArticleIngredientDomainLang" in the same request are not supported.

Example:

```
http://localhost:1512/rest/V1.0/list/Article/ArticleIngredient/byCatalog?
fields=ArticleIngredient.JuiceContentPercentage,ArticleIngredientLang.IngredientState
ment,ArticleIngredientDomainLang.GMOIngredients&
catalog=ECCnet_TEST_CAT
```

Qualified fields can be used in one request:

```
http://localhost:1512/rest/V1.0/list/Article/byCatalog?
fields=ArticleIngredient.JuiceContentPercentage,
ArticleIngredientLang.IngredientStatement(de),ArticleIngredientLang.IngredientState
ment(en),
ArticleIngredientDomainLang.GMOIngredients(de),ArticleIngredientDomainLang.GMOIngredi
ents(en)&
catalog=ECCnet_TEST_CAT
```

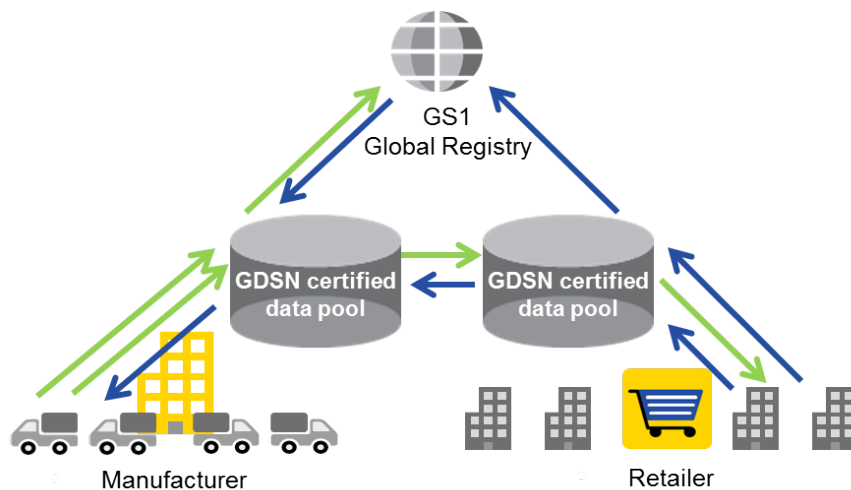
2 GDSN Accelerator

2.1 GDSN Accelerator (Major Release 3)

2.1.1 Providing an interface to the Global Data Synchronization Network (GDSN) for accurate product data exchange

This add-on to Product 360 lets you synchronize product data with GS1 GDSN standards. Throughout the lifecycle of a product, many people rely on access to information about that product and that product information must constantly be maintained and updated by manufacturers and brands. After all, reliable and even more detailed product information is required to comply with changing regulations and satisfy consumer expectations of accessing product information from any channel. Distributors, retailers, and operators are constantly challenged with insufficient and inaccurate product data across their warehouses, stores or online.

Enter the Global Data Synchronization Network (GDSN), a system of interoperable data pools and a global product classification system and registry, that enables companies to exchange standardized and synchronized supply chain data with their trading partners. Suppliers and retailers can cut down the cost of building point-to-point integrations and speed product introductions by accessing the most accurate and most current product information.



The **Informatica GDSN Accelerator** is an add-on to the Informatica Product 360 system that provides access to the 1WorldSync Item Management and Data Sync Engine - two GDSN-certified data pools. It is designed to help organizations securely and continuously exchange, update, and synchronize

product data with trading partners according to the standards defined by GS1.

After being integrated in your Product 360 system, the GDSN accelerator helps you easily transfer product data to the data pool and fully control the information shared with a specific trading partner. The solution offers great flexibility to map item attributes to the GDSN standard and can be tailored to customer specific needs like providing industry-specific or optional fields.

2.1.1.1 There are many benefits of using the GDSN Accelerator:

- Provides improved and consistent product data quality across the entire supply chain, increasing the accuracy of orders, reducing invoice errors, streamlining processes and slashing supply-chain costs
- Improves efficiency and collaboration with trading partners as it shares and updates product information quickly
- Streamlines and accelerates item setup processes and automates data exchange between data sources and recipients by accessing a single source of the truth. This helps speed time to market and time to shelf for new products or products with modified attributes like ingredients or package sizes.
- Can support manufacturers, suppliers and retailers to comply with regulations, and satisfies consumer and regulatory demands for more and better product information
- Informatica is a partner of 1WorldSync, atrify and GS1 Germany

2.1.1.2 Item Management

Item Management "**IM**" is the data pool solution of 1WorldSync inc, former 1Sync. This is mainly used in USA.

2.1.1.3 Data Sync Engine

Data Sync Engine "**DSE**" is the data pool solution of atrify, former 1WorldSync GmbH and former SA2 WorldSync. This pool is more commonly used in Europe.

2.1.2 The GDSN accelerator for Product 360

The GDSN accelerator consists of three main features. The first one is the data model extension. For GDSN Major Release 3, we made our data model compliant and added further parts of the GDSN Extension *Food and Beverage*. The second feature provided with the GDSN accelerator is a lot of data quality rules and rule configurations to check if the data which will be sent to the GDSN data pool is valid. Last but not least it contains templates to export the data in the corresponding format which will be accepted by the GDSN data pool.

2.1.2.1 Data model extension

The Product 360 data model supports the core GDSN data model attributes, most of the Food and Beverage extension and the Canada extension (IM only). A complete list of all provided fields can be found in the chapter "GDSN Accelerator field list". The included languages are English and German for field labels, value lists and descriptions, other languages need to be accessed from the local GS1 Organization, as they refer to standardized terms and cannot be translated without introducing ambiguity and confusion.

Packaging hierarchies

GDSN defines a special hierarchy for items, the so-called packaging hierarchy. In Product 360 this packaging hierarchy is represented by a special reference type called *Next lower level*. You can build a packaging hierarchy by creating a reference of type *Next lower level* from one item to another. The important thing here is that the parent has a reference to the child with the corresponding quantity (number). One example is shown in the screenshot below where a case was dropped on a pallet to create a packaging hierarchy. Here the created packaging hierarchy will contain one pallet with 25 cases.

Drag & Drop event

You have selected "Pallet" of type "Item" as the target.

☒ Create a reference between the item "Case" and the item "Pallet"

Reference type:

Number:

☐ Create a component

The item is added to the selected kit as a static or variable component.

Component type:

Sequence:

Quantity:

OK Cancel

The existing "Item references" view can be used to edit respectively delete the *Next lower level* references:

Pallet

	Reference type	Referenced object type	Referenced item	Referenced object number	Number
1	Next lower level	Item	Case A	Case A	25
2	Next lower level	Item	Case B	Case B	50

1 element selected Reference type ▼

2.1.2.2 New UI components

With the GDSN accelerator Product360 provides new UI views within several UI perspectives like "GDSN (core)" and "Food and beverage (core)" where the attributes can be displayed and maintained. All views except the *GDSN packaging hierarchy* support the full repertoire of standard functionality like grouping, filtering etc. Please find more detailed information for the packaging view in chapter "[GDSN Accelerator operation\(see page 94\)](#)". Product 360 Web also provides UI components to visualize and maintain GDSN data with some limitations mentioned below.

Informatica MDM - Product 360 10.1- Accelerators

The screenshot displays the Informatica PIM 7.1 interface. The top window shows the 'GDSN packaging hierarchy' for item 10000000000017, which is a base unit or each. The bottom window shows the 'Target market specific GDSN data' for item 18, which is 'Coffee Supreme X1D 1000'. The interface includes a sidebar with navigation options like 'Structures', 'Catalogs', 'Media', 'Tasks', 'Queries', and 'Search'. The main area displays a table of GDSN data with columns for Item no., GTIN, EAN UCC code, and Target market. The 'Coffee Supreme X1D 1000' item is highlighted, showing its status as '01 New' and its target market as 'USA'.

Limitations of Product 360 Web:

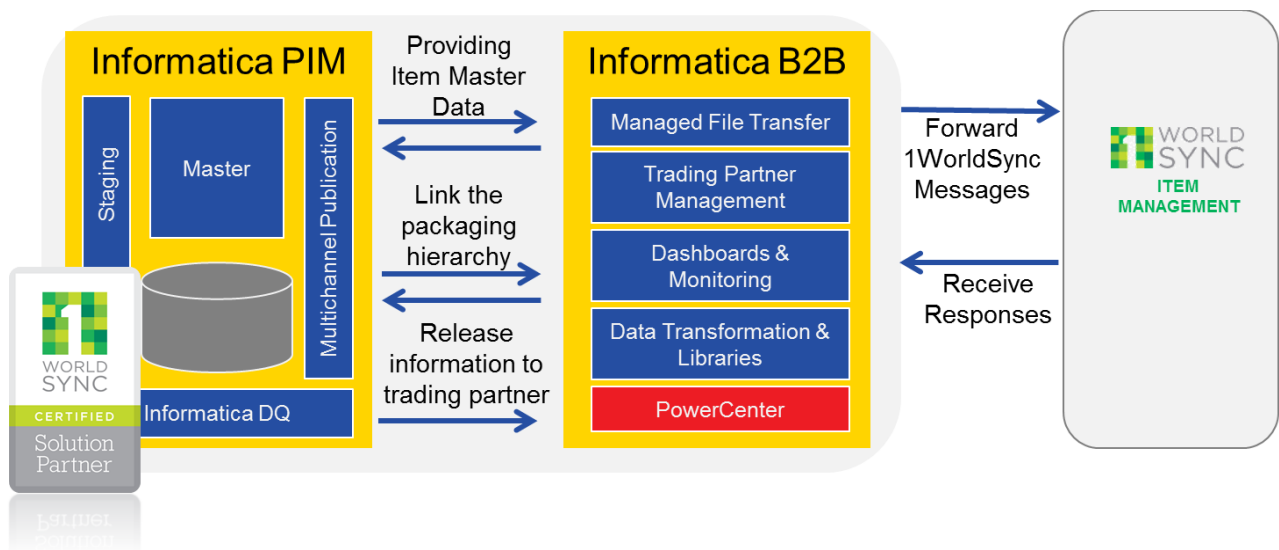
- The publication status view is also not available in Product 360 Web
- In Product 360 Web language specific GDSN attributes can only be maintained in the key language, but it is possible to adjust the XML configurations

2.1.2.3 Communicating with the GDSN data pool using Informatica B2B DX



Please be aware of the approach using OpenAS2 instead of Informatica B2B. See the relevant chapter "[GDSN Accelerator with OpenAS2](#)(see page 404)" in this documentation.

Product 360 uses the core competence of its export to send the data to the GDSN pool. The export creates an XML file according to an XSD schema defined by the pool in use. This file will be sent to Informatica B2B DX which forwards it to the connected GDSN pool. Informatica B2B DX can be seen as a mediator who communicates with the GDSN pool, does some data transformation and writes the answers from the GDSN pool back to Product 360. The picture below gives an overview of the interaction of all three systems.



Answers from Informatica B2B DX and the GDSN pool

After Informatica B2B DX receives a message from Product 360, it immediately writes a "Publication status" back to Product 360 to indicate that the message was received. The message will be forwarded to the GDSN pool and the answer of the pool will be transferred from Informatica B2B DX to Product 360 by writing a "Publication status" via the Product 360 Rest API.

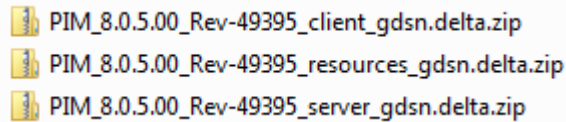
2.1.2.4 Media assets

The data model of media assets was enhanced with five new GDSN related fields. The field "Uniform resource identifier" contains the URL to the image to add to your data in the GDSN pool.

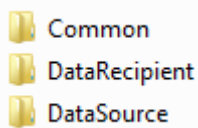
Please note that there is currently no supported way to upload physical images to the GDSN pool.

2.1.3 GDSN Accelerator package content

The GDSN accelerator package comes in the folder PIM_8.0.5.00_GDSN in a zip file called PIM_8.0.5.00_Accelerators.zip and contains the following content.



While the client and server zip contains the GDSN bundles which have to be simply installed via control center the "resources" package contains additional content which is shown below.



2.1.3.1 Selecting the right package

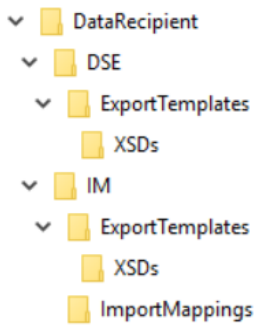
Before the GDSN accelerator is installed, it has to be clear which scenario is applicable. We support two different GDSN data pools, Item Management (IM) and Data Sync Engine (DSE). For both we support the data source as well as the data recipient side which leads to 4 different configuration possibilities. Depending on your scenario the corresponding export templates, import mappings and data quality rules have to be selected.

The "Common" folder

The "Common" folder contains things that are required for all scenarios, for example the GPC structure import which is relevant for the data recipient scenario as well as for the data source scenario and of course for both GDSN data pools.

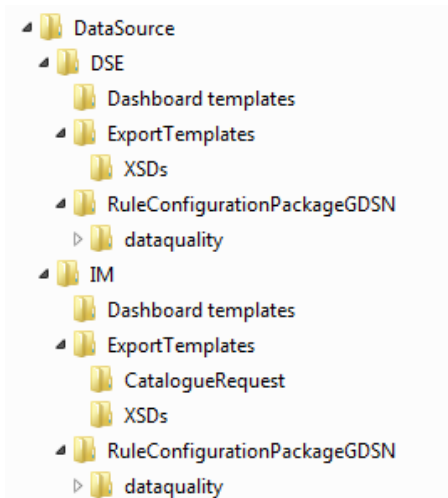
The "DataRecipient" folder

The "DataRecipient" folder contains all things necessary to install the data recipient scenario. This folder is divided again into the subfolders "DSE" and "IM" to differentiate between the GDSN data pools. For the recipient we currently only provide some export templates as well as basic import mapping to be able to import CINs into Product 360.



The "DataSource" folder

The "DataSource" folder contains all things necessary to install the data source scenario. Like the data recipient folder it is divided into a "DSE" and "IM" subfolder which contains the corresponding Dashboard templates, Export templates (with XSD schema files) and rule configurations. The dashboard templates are examples how a dashboard could be configured for a corresponding business user working with GDSN. Due to the fact that the choreography is different between IM and DSE the dashboards are also different. The export templates and the corresponding XSD schema files are also different depending on the GDSN data pool. This has to be considered when importing them. Last but not least there are the rule configurations files. While Product 360 provides a lot of standard GDSN rule configurations which are applicable for DSE and IM data pool, there are also a lot of additional IM specific rule configurations which are located in the IM folder.



2.1.4 GDSN Accelerator installation

2.1.4.1 Prerequisites

B2B Data Exchange environment

The installation of the Informatica B2B Data Exchange - 1SYNC Accelerator is required to use the GDSN Accelerator successfully. For more information, see the corresponding Release Notes available in the

Informatica MySupport portal. Based on the information there, you can find ongoing information in chapter "[B2B Data Exchange](#)(see page 60)".

The receive endpoint directory for Informatica B2B

DX **wf_m_1Sync_message_from_ds_to_sdp** is called **<B2B_DS_INPUT_PATH>** hereafter.

Product 360 environment

The version of the installed Product 360 must be 8.0.5 or higher. Also the standard Data Quality Rules and reference data (contained in the DataSource folder of the file **PIM_<Version>_Rev-xxxxx_resources_gdsn.delta.zip**) have to be installed.

2.1.4.2 Installing the GDSN Accelerator

Download GDSN Accelerator zip

The **GDSN Accelerator** is available via Informatica Shipping. It consists of the folder **PIM_<Version>_GDSN** in the file **PIM_<Version>_Accelerators.zip**.

Extract the GDSN Accelerator zip

Unpack the file **PIM_<Version>_Accelerators.zip** to a temporary directory on the Product 360 environment. It contains the following artefacts:

- **PIM_<Version>_Rev-xxxxx_client_gdsn.delta.zip**
 - contains PIM Desktop feature and plugins
- **PIM_<Version>_Rev-xxxxx_resources_gdsn.delta.zip**
 - contains export templates, import mappings, DQ rules and rule configurations, Product 360 Web view definitions
- **PIM_<Version>_Rev-xxxxx_server_gdsn.delta.zip**
 - contains PIM Desktop feature and plugins

Product 360 environment configuration

Unpack the archive **PIM_<Version>_Rev-xxxxx_resources_gdsn.delta.zip** into a temporary folder **<GDSN_RESOURCES_PATH>**.

Product 360 Server

1. Unpack the archive **PIM_<Version>_Rev-xxxxx_server_gdsn.delta.zip** into the Product 360 server folder or use the PIM Control Center.

Install GDSN Data Quality Rule Configurations

1. Backup the Product 360 server's **dataquality** folder (PIM 8.0: **<PIM_SERVER_SHARED_DIR>/dataquality**).
2. Copy (and replace) the file **StandardDataQualityMappingProfile.xml** to the subfolder **config** of the Product 360 server's **dataquality** folder. (**<PIM_SERVER_SHARED_DIR>/dataquality/config**)
 - a. For IM the StandardDataQualityMappingProfile.xml to be used can be found here:
 \DataSource\IM\RuleConfigurationPackageGDSN\dataquality\config

- b. For DSE the StandardDataQualityMappingProfile.xml to be used can be found here:
\DataSource\DSE\RuleConfigurationPackageGDSN\dataquality\config

i If there are problems during startup due to same objects (because of same identifiers) used in both configuration sets, the server won't start. The logs give detailed information about conflicting objects.

Later after the complete GDSN accelerator installation/migration, execute DQ runs to get a status for the item for each of the new rule configurations. And the channels status is restored as well.

Adjust Repository

It is not necessary to do repository adjustments for IM. During the start of the Product 360 server the repository will be adjusted automatically according to the configured GDSN pool.

Those steps are described in the "[Repository configurations\(see page 64\)](#)" chapter as well as options for manual custom-specific repository adjustments.

If you are using the DSE pool, please have a look at "[Configuration for DSE\(see page 60\)](#)", to see the necessary adjustments.

Adjust application modules

In the "application_modules.properties" file you have to activate the used GDSN modules, see chapter "[Application modules\(see page 59\)](#)" for more details.

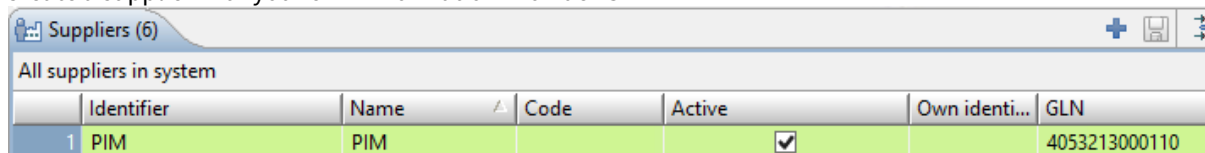
Product 360 Desktop Client

Install Product 360 Desktop Client enhancements

1. Unpack the archive **PIM_<Version>_Rev-xxxxx_client_gdsn.delta.zip** into the Product 360 client folder.
2. Start the client.

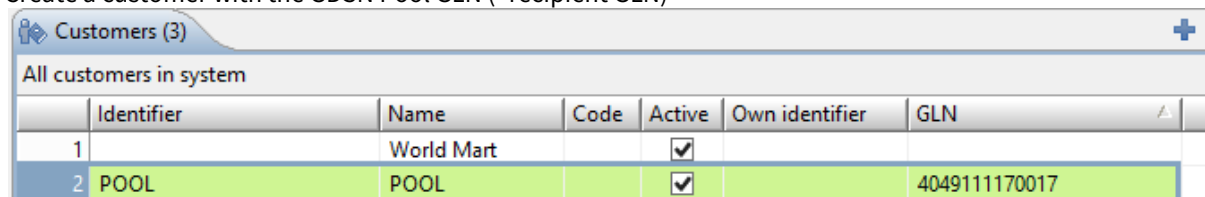
Create supplier and customer

1. Create a supplier with your **own** Information Provider GLN



Suppliers (6)						
All suppliers in system						
	Identifier	Name	Code	Active	Own identi...	GLN
1	PIM	PIM		<input checked="" type="checkbox"/>		4053213000110

2. Create a customer with your own Information Provider GLN (this is needed for publication to market groups(IM Only))
3. Create a customer with the GDSN Pool GLN (=recipient GLN)



Customers (3)						
All customers in system						
	Identifier	Name	Code	Active	Own identifier	GLN
1		World Mart		<input checked="" type="checkbox"/>		
2	POOL	POOL		<input checked="" type="checkbox"/>		4049111170017

4. Maintain the GLN on all your customers

The GLNs are necessary to get the answers from the pool back to the PIM and are used by the B2B to create the publication status messages.

Load and save export templates

See ["Export templates"](#)(see page 69) for instructions.

Install GDSN Data Quality rules and reference data

1. Add GDSN specific rules:
 - a. Open the perspective "Data quality"
 - b. Select any custom rule configuration
 - c. Open the "Select data quality rule" dialog (".." -button) (If the rule configurations are read only, create new one to be able to import rules.)
 - d. Click the button "Add rules from file"
 - e. choose the file **<GDSN_RESOURCES_PATH>/Common/RulePackageGDSN/Informatica_PIM_GDSN.xml**
 - f. Wait until the rules were added successfully
2. Add GDSN specific reference data
 - a. In the "Select data quality rule" dialog (see step 2.c. above) click the button "Add reference data from file"
 - b. Choose the file **<GDSN_RESOURCES_PATH>/Common/RulePackageGDSN/Informatica_PIM_GDSN.zip**
 - c. in the "Adding reference data" dialog leave the defaults and click "OK".
 - d. Wait, until all dictionaries are deployed, which is done automatically within a server job of type "Deployment of reference data (scheduled)".

Import GPC Structure

1. Download the latest files from the GS1 website: <http://www.gs1.org/gpc/production>
 - a. It is recommended to download the **Combined Published Schema, or the subset of the required industry**
 - b. The excel file **GS1 Combined Published_Schema as at 0000000.xlsx** from the zip archive is needed
2. Create a new **output** structure system with identifier "GPC". Configure that multiple assignments are not possible for this structure system.
3. Use the import mapping **<GDSN_RESOURCES_PATH>/Common/GPC structure import/GPC en.him** to import the downloaded excel file
4. Switch the structure system to type "classification system"

Load Dashboard

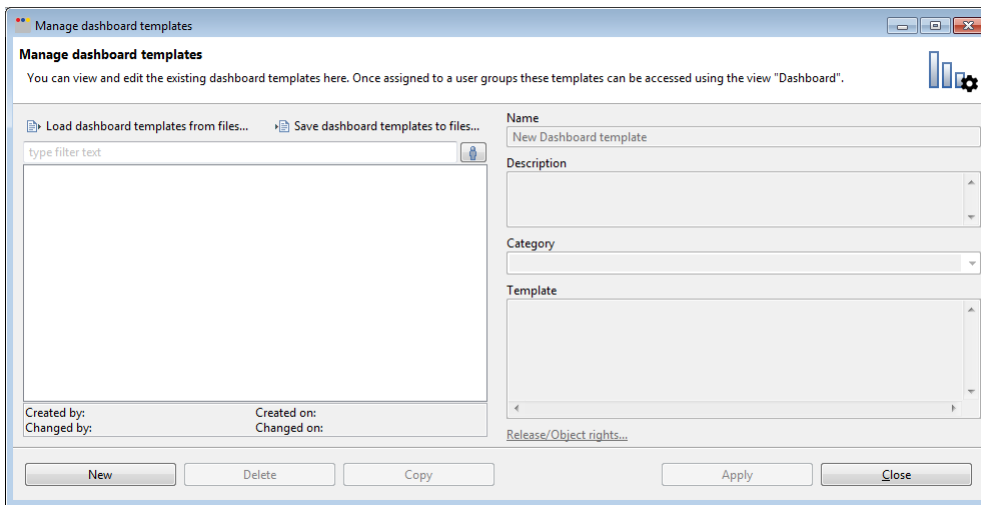
The GDSN accelerator package contains 2 dashboard templates, one for IM data source scenario and one for DSE data source scenario. The templates are designed for business users who are working with GDSN and contain typical widgets for their daily work such as a widget showing how many items are already sent to the pool or how many items are already published.

To use the dashboards please open the "Manage Dashboard templates" dialog at the menu "Management". Then click on "Load dashboard templates from file" and select your dashboard. You can assign this dashboard to your GDSN user group in the organization perspective (in the view "User groups"), so it will be opened whenever a user of this user group opens the web client.

Some parameters need to be adjusted to the company's details:

```
<parameter key="groupingField" value="Article.Status->PublicationStatusEntry.ResponseType('Super CPG','World Mart',DE,PUBLICATION_RESPONSE,{Default})"/>
```

- 'Super CPG' should be replaced with the Name of the Supplier holding the GLN of the own company
- 'World Mart' should be replaced with the Name of the default customer who should be visualized
- DE should be replaced with the required Target Market



2.1.4.3 Application modules

With Product 360 8.0.5 the application module concept was introduced. A module can be seen as business functionality which can be enabled or disabled. Therefore a new property file called "application_modules.properties" located in the configuration folder of the server was introduced. Currently it is possible to enable/disable different GDSN scenarios and the Food and beverage module. When activated, the corresponding perspectives, views and other things like reports and export datatypes are available for the user to import/maintain/export food and beverage or GDSN data.

By default "GDSN" as well as "FoodAndBeverage" are set to "false" and must be activated during the accelerator installation (food and beverage only if needed).

The Food and Beverage module can be used independently of GDSN.

An example for the application_modules.properties file is given in the screenshot below. Please see the following chapters for a specific "DSE" or "IM" configuration.

```
# Defines if the GDSN extension is installed (true) or not (false)
GDSN = true

# Defines if the GDSN pool "IM" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_im = false


# Defines if the GDSN pool "DSE" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_dse = true

# Defines if the GDSN extension is used in "data source" mode (true) or not (false) if installed.
gdsn_data_source = true

# Defines if the GDSN extension is used in "data recipient" mode (true) or not (false) if installed
gdsn_data_recipient = false

# Defines if the food and beverage module is activated (true) or deactivated (false).
FoodAndBeverage = true
```

Please note that both, running multiple pools at the same time and activating more than one role (dataSource and dataRecipient) is not supported.

 Please also note that if GDSN should be used, the corresponding GDSN plugins need to be installed in the client and server.

Configuration for DSE

The following settings are needed in the `application_modules.properties` file for GDSN pool "DSE", data source scenario. If the data recipient scenario should be installed, "gdsn_data_source" has to be set to "false" and "gdsn_data_recipient" has to be set to "true". Again, having both scenarios active at the same time is not supported.

```
# Defines if the GDSN extension is installed (true) or not (false)
GDSN = true

# Defines if the GDSN pool "IM" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_im = false

# Defines if the GDSN pool "DSE" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_dse = true

# Defines if the GDSN extension is used in "data source" mode (true) or not (false) if installed.
gdsn_data_source = true

# Defines if the GDSN extension is used in "data recipient" mode (true) or not (false) if installed
gdsn_data_recipient = false

# Defines if the food and beverage module is activated (true) or deactivated (false).
FoodAndBeverage = true
```

Configuration for IM

The following settings are needed in the `application_modules.properties` file for the GDSN pool "IM", data source scenario. If the data recipient scenario should be installed, "gdsn_data_source" has to be set to "false" and "gdsn_data_recipient" has to be set to "true". Again, having both scenarios active at the same time is not supported.

```
# Defines if the GDSN extension is installed (true) or not (false)
GDSN = true

# Defines if the GDSN pool "IM" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_im = true

# Defines if the GDSN pool "DSE" is used (true) or not (false) if the GDSN extension is installed.
gdsn_pool_dse = false

# Defines if the GDSN extension is used in "data source" mode (true) or not (false) if installed.
gdsn_data_source = true

# Defines if the GDSN extension is used in "data recipient" mode (true) or not (false) if installed
gdsn_data_recipient = false

# Defines if the food and beverage module is activated (true) or deactivated (false).
FoodAndBeverage = true
```

2.1.4.4 B2B Data Exchange

- [Introduction](#)(see page 61)
 - [B2B Data Exchange GDSN Accelerator Components](#)(see page 61)
 - [Item Management \(IM\) & Data Sync Engine \(DSE\)](#)(see page 62)
- [Prerequisites](#)(see page 62)
- [Installation](#)(see page 62)

- [B2B Data Exchange GDSN Accelerator configuration](#)(see page 63)
 - [DX Endpoints](#)(see page 63)
 - [MFT](#)(see page 64)



Please be aware of the approach using OpenAS2 instead of Informatica B2B. See the relevant chapter "[GDSN Accelerator with OpenAS2](#)(see page 404)" in this documentation.

Introduction

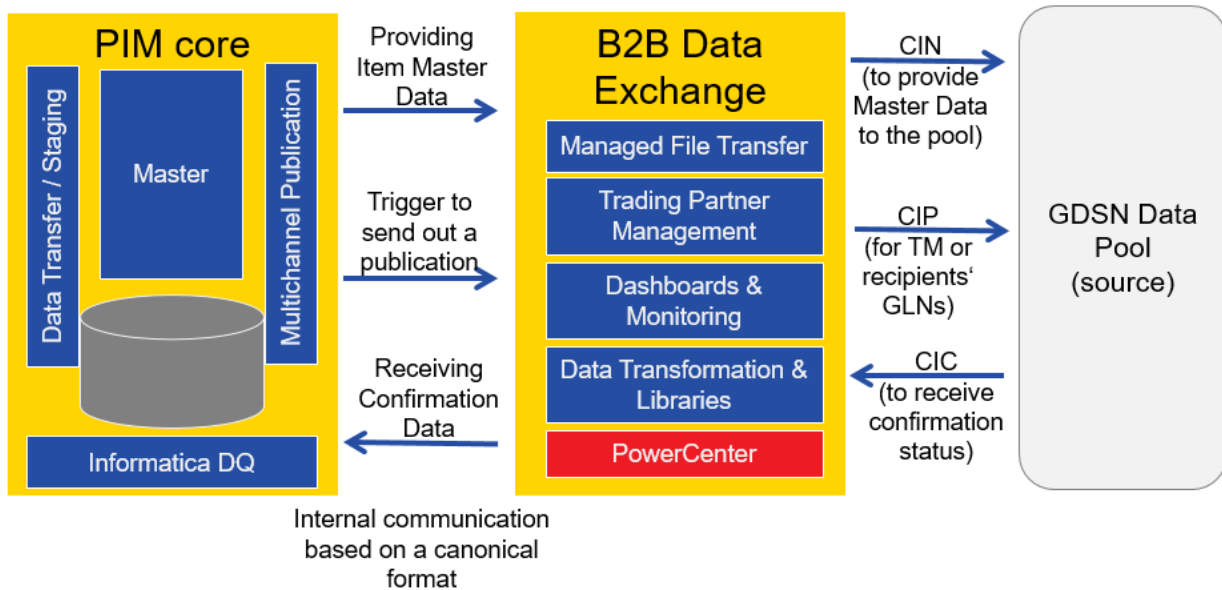
B2B Data Exchange GDSN Accelerator provides a packaged B2B Data Exchange solution that addresses common business-to-business GDSN integration use cases for vertical industries and reduces implementation efforts. It processes the exchanged messages, mediates communication between GDSN data pools and partners and tracks the correlation between messages sent to the data pool and the responses/acknowledgments received from the data pool through the reconciliation status.

B2B Data Exchange GDSN Accelerator Components

B2B Data Exchange GDSN Accelerator comprises of several Informatica products:

- Managed File Transfer (MFT) - connect to GDSN data pool using AS2/SFTP protocols
- Dashboards and Monitoring - manage and monitor data communication using B2B Data Exchange Operator Console.
- Data Transformation (DT) - transforms data from one format to another. In the GDSN context, DT handles the transformation between the Product 360 format and the GDSN data pool message format.
- PowerCenter - Integration layer with workflows containing DT services, B2B Data Exchange transformations, request-response data correlation, and more.

The image below shows the architecture data flow of the B2B Data Exchange GDSN Accelerator in the Data Source scenario.



Item Management (IM) & Data Sync Engine (DSE)

Product 360 can exchange product data information with two different GDSN data pools - IM and DSE. These data pools handle different data formats, which is the basis of how the data is being represented and exchanged between Product 360 and GDSN.

This difference means that while most of the B2B Data Exchange GDSN Accelerator components remain the same, the DT services are different when working with IM and DSE data pool.

Prerequisites

Before you install the B2B Data Exchange GDSN Accelerator, ensure that you have a valid license with the GDSN options.

Ensure that you have installed the following products:

- PowerCenter
- Data Transformation
- B2B Data Exchange
- MFT (In case the customer has its own MFT solution, then Informatica MFT is not required.)

Note: Informatica product version may vary. The obtained B2B Data Exchange GDSN Accelerator should be compatible with whatever Informatica product version you installed.

Installation

Configure Informatica application services based on the products and functionality that are required to deploy the GDSN Accelerator. For example, if you use PowerCenter to run the B2B Data Exchange GDSN Accelerator, you need to configure the following:

1. PowerCenterRepositoryService

2. PowerCenter Integration Service
3. Web Services Hub Service

Install and configure the different components of the B2B Data Exchange GDSN Accelerator - Follow the instructions as described in the Release Note document, which is included in the B2B Data Exchange GDSN Accelerator package.

B2B Data Exchange GDSN Accelerator configuration

DX Endpoints

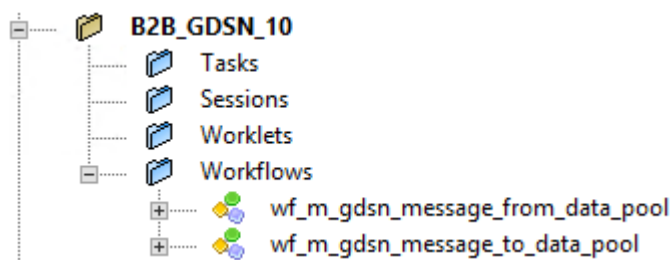
Configure the DX endpoints to ensure correct data exchange between the different products that are involved in the GDSN solution.

The image below is from B2B Data Exchange Operator Console and it shows the DX Endpoints that are part of the GDSN Accelerator that needs to be configured.

Endpoints		
Find	<input type="text"/>	<input type="button" value="Search"/>
Name ▲	Type	Partner
endpoint_gdsn_message_from_data_pool	File receive	Data Pool (GDSN)
endpoint_gdsn_message_from_dx_to_data_pool	File send	Data Pool (GDSN)
endpoint_gdsn_message_from_dx_to_PIM	File send	PIM
endpoint_gdsn_message_from_PIM_to_data_pool	File receive	PIM

For example, the DX endpoint "endpoint_gdsn_message_from_PIM_to_data_pool" specifies where B2B Data Exchange will receive the files that were exported from Product 360. Then, according to the B2B Data Exchange GDSN Accelerator specification, since the format of the exported file from Product 360 doesn't comply with the data pool expected format, B2B Data Exchange sends the file to a PowerCenter workflow for further processing and data transformation.

The image below is from the WorkflowManager client tool and it shows the PowerCenter workflows that are part of the GDSN Accelerator.



For example, the PowerCenter workflow "wf_m_gdsn_message_to_data_pool" execute set of tasks to process the exported file from Product 360 before it's sent to the data pool.

Within the WorkflowManager you specify the JMS, JNDI and Http connections that are required for the GDSN Accelerator to work. For more information, please refer to the Release Notes documentation that is part of the GDSN package.

MFT

In case the customer wants to use Informatica MFT product for data communication with GDSN data pool, then a set of steps are required to be performed in MFT and B2B Data Exchange.

2.1.4.5 Repository configurations

Repository auto-adjustments

During the start of the Product 360 server the repository will be adjusted automatically according to the configured GDSN pool. The following steps will be done:

Activate GDSN repository entities

The GDSN modules are implemented as second-level entities below the "Item" root entity. In this context, the terms "GDSN modules" and "GDSN specific repository entities" are used interchangeably.

All GDSN specific repository entities are deactivated by default in the repository. If GDSN is installed and activated (see "[Application modules\(see page 59\)](#)"), all GDSN repository entities will be enabled during server start.

Note: The "Active" attribute of the repository entities is the only one that is changed here. All other attributes - like name, import purpose, ... - as well as the fields and the sub-entities can be configured in the repository as usual.

In addition to that, there're few data fields that are pool-specific. They will be disabled during the server start according to the configured GDSN pool.

Attention: It is strongly recommended that you don't change the "scope" settings of repository fields.

Activate enumeration entries of GDSN enumerations

The entries of GDSN enumerations defined in the "EnumFragment.repository" file will be added to the enumerations.

There're some enumeration entries that are pool-specific. That's why some enumeration entries got an additional "scope" parameter that is analysed during server startup and ensures consistent valid value lists. Not applicable enumeration entries will not be added.

Attention: It is strongly recommended that you don't change the "scope" settings of enumeration entries.

"Next lower level" item reference type

The GDSN packaging hierarchy is implemented by an additional item reference type "Next lower level". That reference type will be added to the "Enum.Article.Article.ReferenceTypes" repository enumeration, the default value is "12000". In addition to that, an enum param "GDSN.packagingHierarchy.reference" is added to that enumeration, it is needed by the business layer, e.g. to provide the hierarchy view with appropriate data.

Manual repository adjustments

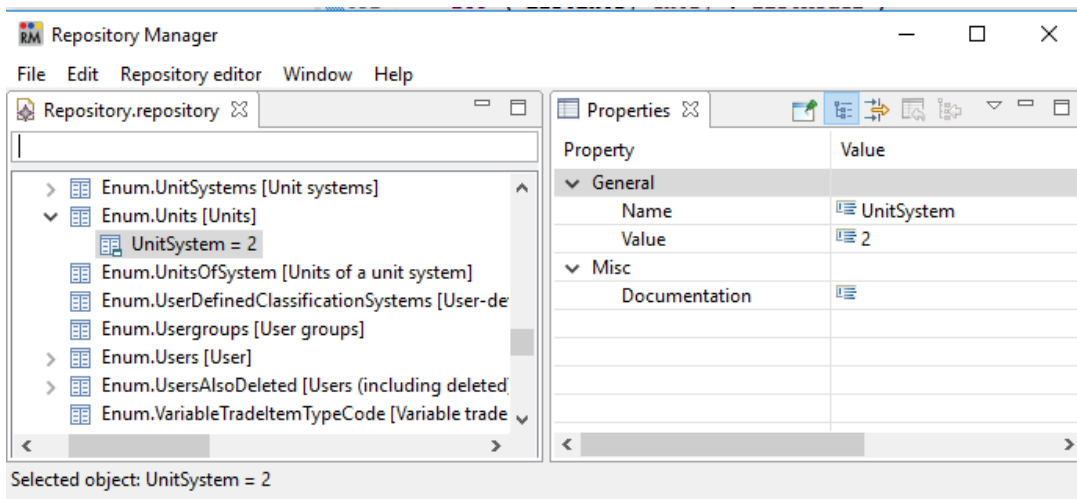
Depending on your needs, you need to make some settings manually.

i For all GDSN specific enumerations (e.g. Enum.PreparationType) the standard repository file (repository.repository) only contains header information. The enumeration entries itself can be found in the repository fragment file **EnumFragment.repository**. This file is located in **<PIM_CONTROL_CENTER_INSTALLATION_FOLDER>/configuration/HPM/gdsn**. It can be opened with the Repository Manager similar to the standard repository file.

Set GDSN unit system as default unit system

By default, the unit system "System units" (2) is used by the "Units" repository enumeration. All GDSN UOM fields have got own unit enumerations according to the GDSN specification.

If you want non-GDSN fields to use only GDSN units you should switch the default unit system to "70".

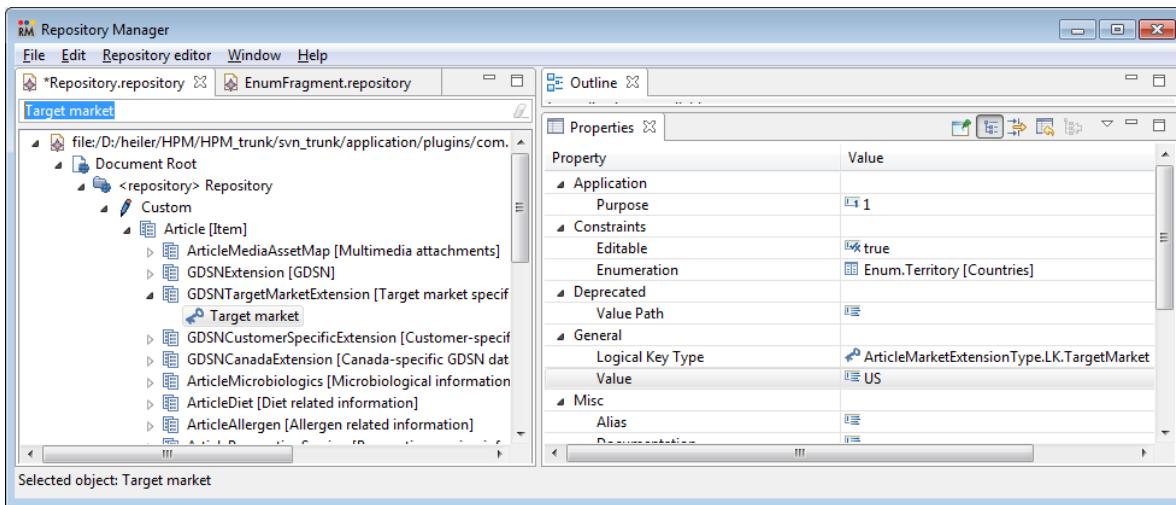


Change default field qualification

In order to work more efficiently, you should set the default values of some logical keys to your needed value. Note that you don't have to set the values of non-editable or non-visible logical keys and fields; for editable fields it's sufficient to set the logical key value.

Target market

There's a lot of target market specific fields. If you want to mainly use one dedicated target market you should set the default value of the corresponding logical keys and fields to that target market code.



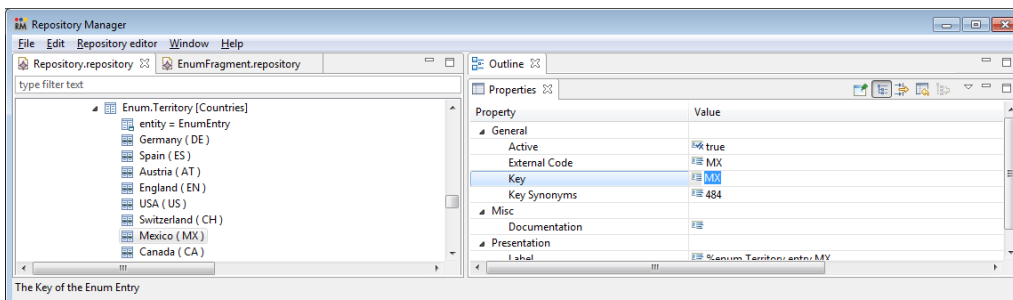
UOM type

The default uom type value is "metric" (METRIC). If you maintain imperial unit of measures primarily you should change the default value to "IMPERIAL":

Additional enumeration entries

Target markets

If needed, enhance the enumeration **Countries** (Enum.Territory) with further target markets.



For each target market, also add an entry in the server's plugin_customization.ini like the following:

com.heiler.ppm.gdsn.core/<pool code>.Enum.Territory.<enum_key> = <TM_key> (where <pool code> is 1WS (for IM) or GDSN (for DSE), <enum_key> is the key of the enum entry, <TM_key> is the pool specific key of the target market)

IM specific target market entries in plugin_customization.ini

```
# mexico
com.heiler.ppm.gdsn.core/1WS.Enum.Territory.MX = MX
# canada
com.heiler.ppm.gdsn.core/1WS.Enum.Territory.CA = CA
```

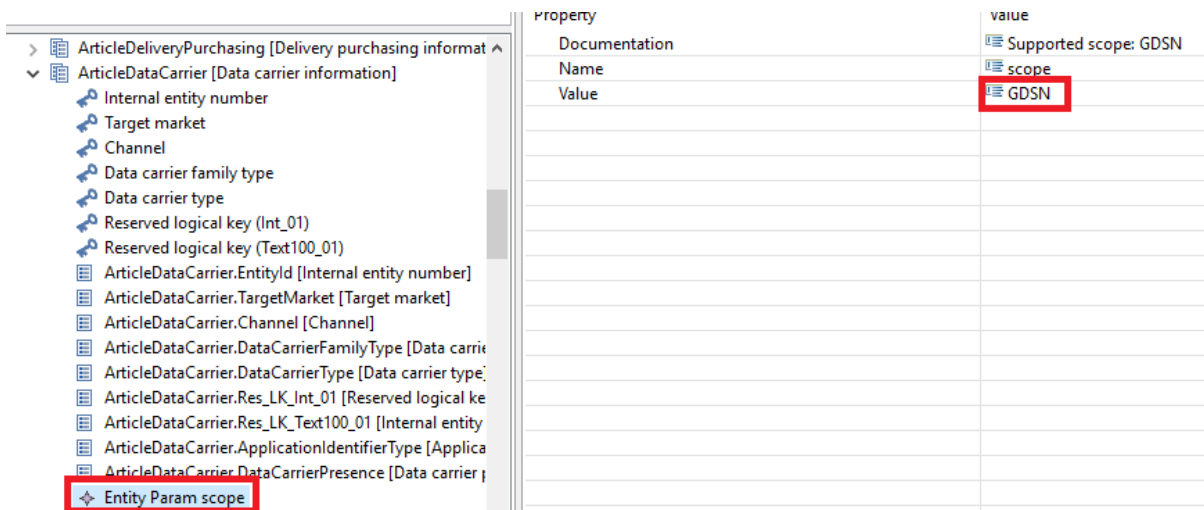
DSE specific target market entries in plugin_customization.ini

```
# mexico
com.heiler.ppm.gdsn.core/GDSN.Enum.Territory.MX = 484
# canada
com.heiler.ppm.gdsn.core/GDSN.Enum.Territory.CA = 124
```

Customizing

Hide a GDSN module

When you do not want to deal with an entity which is not needed for your scenario, it is possible to deactivate the entity in the Repository Manager. Select the unwanted entity and edit the "Entity Param scope" of the entity.



When deleting the value GDSN those entities won't be activated during server start anymore. Furthermore, the corresponding views are not available in the UI.

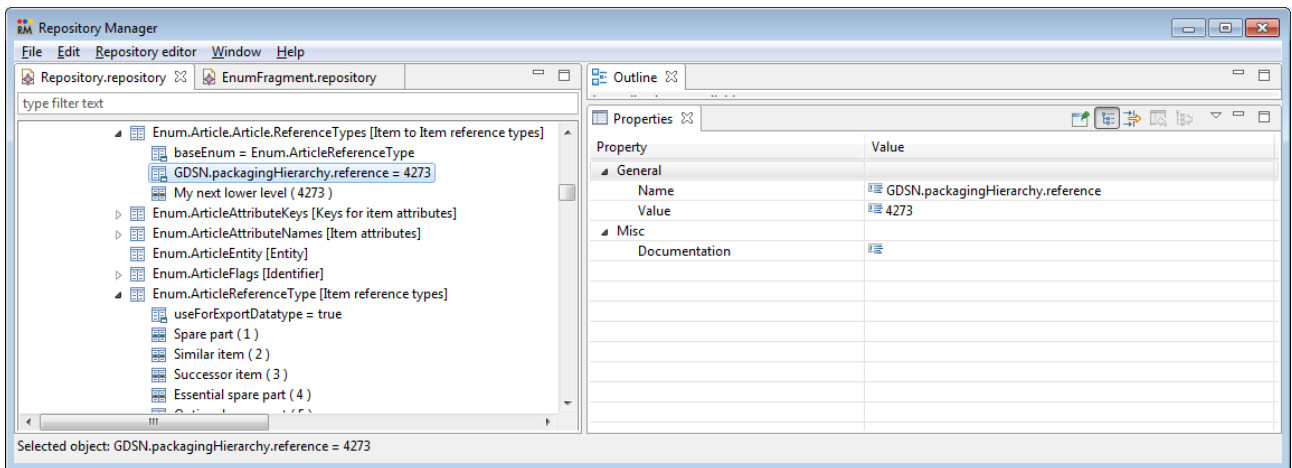
Take care when deactivating an entity because all **references** to the entity in other Product 360 components have to be deleted manually.

Example for these components are:

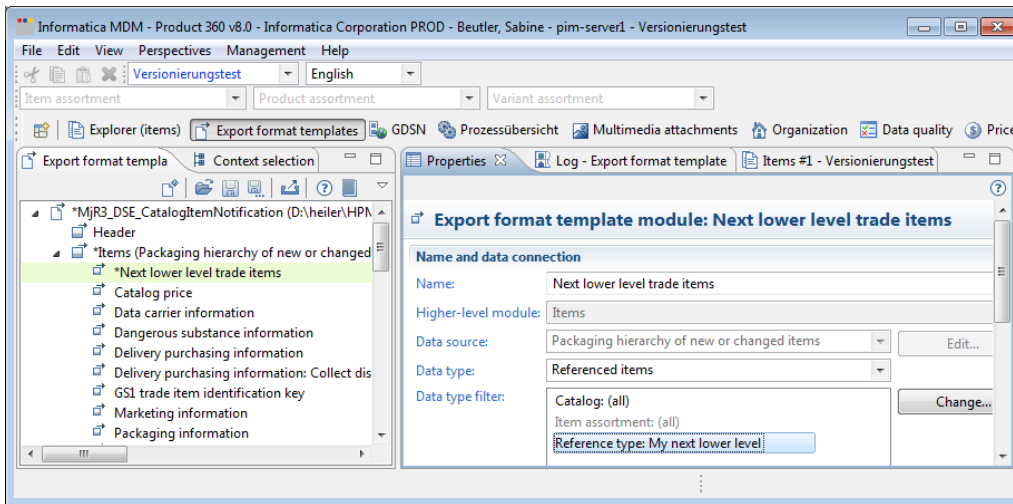
- Export templates
- Import mapping
- Custom article detail XML files
- DQ rule configurations
- Product 360 customizations

Change the "Next lower level" reference type

If you want to use another reference type for the packaging hierarchy of items, you have to ensure this reference type is part of the "Enum.Article.Article.ReferenceTypes" repository enumeration. Furthermore, you have to add an enumeration parameter "GDSN.packagingHierarchy.reference" with your reference type as value to that enumeration.



In case you're using DSE you have to adjust the CatalogItemNotification export template accordingly:



Configuration for DSE

There're some additional repository changes needed if you want to use the DSE GDSN pool.

Adjust length of repository fields

Field identifier	Field length (repository)	Field length (DSE)
GDSNTargetMarketExtensionLang.FunctionalName	70	35
GDSNTargetMarketExtensionLang.Variant	70	35
Certificate.Value	200	120

Please note: Although this could be also done automatically during startup we decided to leave it as a manual step. This has mainly transparency reasons but also has the advantage that field length can still be changed by Professional Services, e.g. when the data model is used in another context than GDSN.

2.1.5 GDSN Accelerator setup

After installing the GDSN accelerator you should do some steps to setup the Product 360 working environment, like arrange suitable perspectives if needed, configure GDSN export templates, setup jobs to automatically create and send GDSN messages.

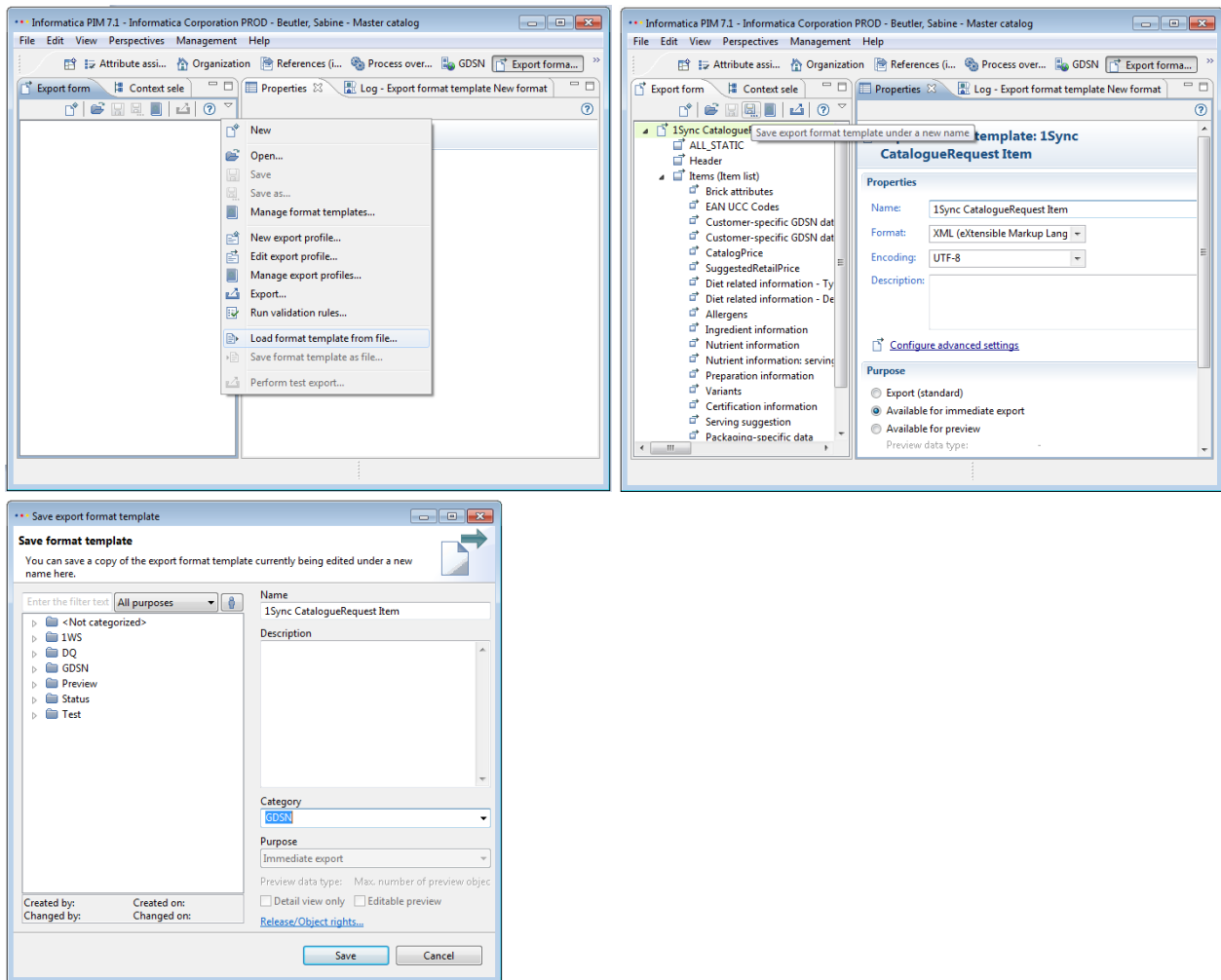
2.1.5.1 Export templates

Note: If you want to export to the DSE pool adjust the repository like described in "[Configuration for DSE](#)(see [page 68](#))".

Load and save the export templates

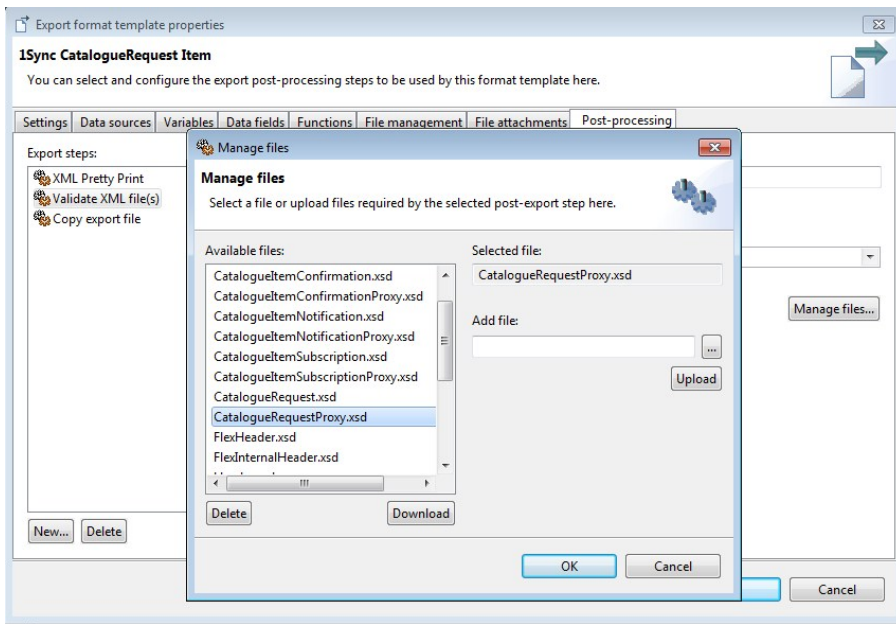
The GDSN accelerator package contains various export template files that you should load and then save within Product 360. This is necessary to be able to configure the export templates, to schedule repeating export jobs as well as to create one-click exports for immediate data transfer to the GDSN pool. Depending on the GDSN data pool the export templates are completely different. In the subchapters it is described which are needed for the corresponding IM or DSE data pool.

Open the "Export format templates" perspective, choose "Load format template from file...", select one export format template file, assign or create an appropriate category, for example "GDSN", and save the template into Product 360 and repeat those steps for all export format templates:



Add XSD schema files

After saving the export templates, you need to make sure that the needed XSD files are already uploaded to the Product 360 system. Therefore go to the "Post-processing" tab in the property dialog of any export format template and select "Validate XML file(s)" export step. Press on "Manage files..." and examine if the delivered XSD files (e.g. CatalogueRequestProxy.xsd) are already in the "Available files" list. In case those are not uploaded yet, find the XSD files to upload in the GDSN accelerator package and upload them here. Again it is important to upload the XSD files which are appropriate for your GDSN data pool, locations can be found in the subchapters for IM and DSE.



The upload of the XSD files has only to be done once. They will be available for all export format templates.

Export templates for DSE

All important export templates for the Data Sync Engine (DSE) can be found within the DSE folders of the GDSN accelerator package. There are different export templates for the data recipient and the data source scenario, see the chapters below for more details.

- ▼ PIM_8.0.5.05.00_Rev-55577_resources_gdsn.delta
 - > Common
 - ▼ DataRecipient
 - ▼ DSE
 - ▼ ExportTemplates
 - XSDs
 - ▼ IM
 - ▼ ExportTemplates
 - XSDs
 - ImportMappings
 - ▼ DataSource
 - ▼ DSE
 - Dashboard templates
 - ▼ ExportTemplates
 - XSDs
 - > RuleConfigurationPackageGDSN
 - > IM

Data source export templates

These are the following data source export templates:

- Catalog Item Notification (CIN_CatalogItemNotification.ext)
- Catalog Item Publication (CIP_CatalogItemPublication.ext)
- Catalog Item Publication Withdrawal (CIPHW_CatalogItemPublicationHierarchyWithdrawal.ext)

Catalog Item Notification

The *Catalog Item Notification* export template outputs a "Catalog Item Notification" GDSN message containing items. Depending on the configuration, you can transfer new and/or modified trade item information with all detailed data maintained.

More information about provided data fields can be found in chapter "[GDSN Accelerator field list](#)(see page 103)".

Available Operations:

ADD: Adds the item to the pool (can also be used to update the current item). This is the operation that should be used for the automated *Catalog Item Notification* export job.

CORRECT: Is used to update the values for the current item in the pool. It can be used for manual triggered *Catalog Item Notification* exports.

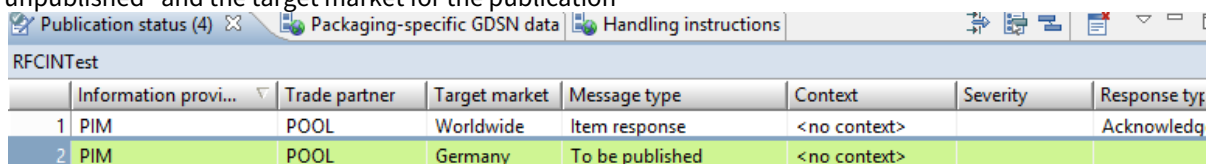
CHANGE_BY_REFRESH: Updates the current hierarchy for the item in the pool if there is no active publication. This operation should only be used for dedicated item hierarchies that are to be transferred to the GDSN pool by manual started *Catalog Item Notification* exports.

Catalog Item Publication

The *Catalog Item Publication* export template creates a GDSN message containing publications for all given items. Depending on the publication status of the items, you can add or delete publications. Furthermore, the message can be sent to a target market or for all GLNs an item is assigned to. You can find more information about the publication in the corresponding "[Publication](#)"(see page 101) chapter.

A publication will be sent if the items to publish follow these conditions:

- The items have been sent to the pool already, there's an Item response publication status for each item
- All items which should be published have to have a publication status entry with "To be published" or "To be unpublished" and the target market for the publication



	Information provi...	Trade partner	Target market	Message type	Context	Severity	Response typ
1	PIM	POOL	Worldwide	Item response	<no context>		Acknowledg
2	PIM	POOL	Germany	To be published	<no context>		

- The configured target market in the export format template corresponds to the target market of the item in "Target market specific GDSN data"
- If you want to publish to specific customers, you have to add a valid trade partner with a valid GLN.

Note: If there is the "<Public>" customer specified instead of a valid trade partner for the publication of the item it will be published to the whole target market taken from the parameter "Target market".

Catalog Item Publication Withdrawal

The *Catalog Item Publication Withdrawal* export template is used to manually modify a publication of single items for a given GLN. Depending on the configuration, you can delete the active GLN publication on an item or change the hierarchy within the item.

A publication withdrawal will be sent if the item follows these conditions:

- The configured target market in the export format template corresponds to the target market of the item in "Target market specific GDSN data".

Available Operations:

PUBLICATION WITHDRAWAL: Used when deleting an active publication.

HIERARCHY_LINK_CORRECTION: Used if you want to change the hierarchy of the specified item. After sending this operation, you can send your item with the changed hierarchy information with a Catalog Item Notification. The publication will be automatically resumed when the new Catalog Item Notification was sent.

Data recipient export templates

These export templates can only be used if you are a data recipient. To change Product 360 to the data recipient mode have a look at the chapter "[Application modules\(see page 59\)](#)".

- Catalog Item Subscription (CIS_CatalogItemSubscription.ext)
- Request For Catalog Item (RFCIN_RequestForCatalogItem.ext)
- Catalog Item Confirmation (CIC_CatalogItemConfirmation.ext)

Export template parameters

Export template parameters can be parameters of data sources, parameters of export post steps or variables. Furthermore variables can be used as parameters of data sources or export post steps.

There are a lot of parameters you can configure in the export templates. All of them can be assigned to one of the following two categories:

Identify

Each of the export templates has two mandatory parameters needed for the communication with the GDSN pool. It is recommended to set the correct values and make those parameters invisible.

- "Information provider" and "Information provider GLN" respectively
This is the supplier representing your own GLN.
- "Recipient" and "Recipient GLN" respectively
This is the customer representing the pool GLN.

Both values have to exist in PIM, otherwise the responses from the pool will not be written back to PIM. Make sure you created the values like described in the section about the creation of supplier and customer.

Select data

Most parameters are used to qualify which data should be exported.

First of all, there is a catalog and an item assortment to determine which items are to be exported.

In addition to that, the target market, a language, a price type, etc. specify which details of items are to be exported.

Export data source configuration

Most GDSN export templates use a form of the [hierarchy of new and changed items data provider](#)(see page 82) as data source. If you want to schedule repeated export jobs, you usually want to send only new and changed items to the pool.

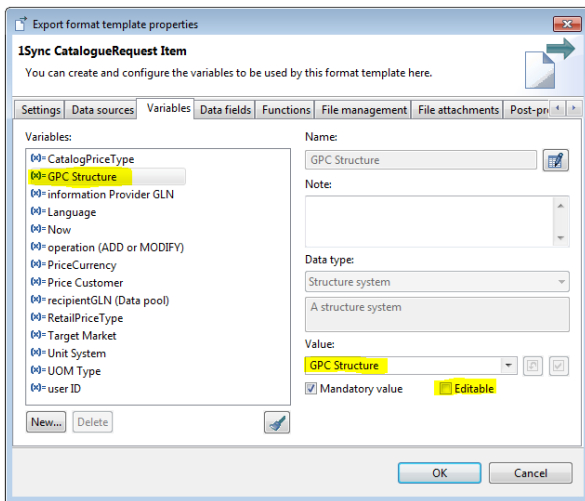
Setting	Value	Editable
Catalog	Master catalog	true
Assortment	<empty>	true
Currency	Euro	true
Reference date	<i>Variable "ReferenceDate", see next chapter</i>	true
Type of change	New and changed items	false
Update assortment	empty	true
GPC structure	GPC	false
Version	Working version	false

You can specify appropriate values for the visible parameters in the export profiles you create of them.

It is recommended to set the "Update assortment" value to "true" for scheduled export jobs, but if you want to start a single export it may be not necessary to update the assortment.

Parameters

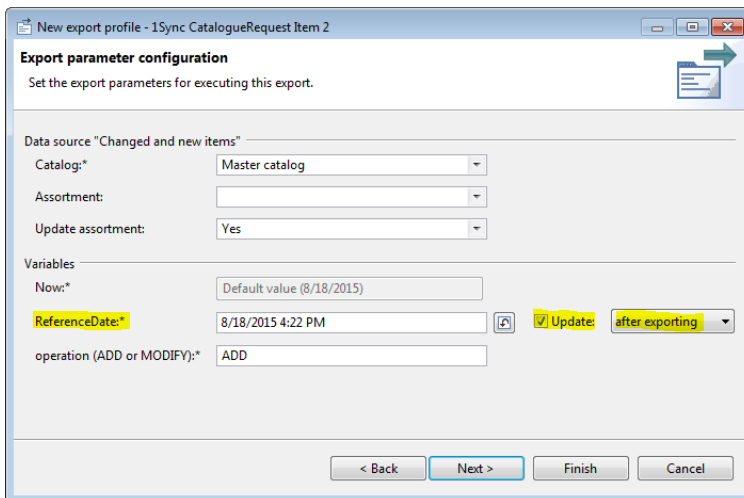
Most parameters of the export template have to have fixed values. You should set those values to the export templates and switch the corresponding parameters to "Not editable" to hide them for further use.



Reference date

If you want a [scheduled export job](#)(see [page 89](#)) to export all items that have been created or updated after the last run, you can use the variable "Reference date" that gets updated with the corresponding date and time for every export job run.

The important step is to select "Update" and "after exporting" for the variable.



XSD schema files

For DSE all available XSD schema files for all GDSN modules are delivered although only some of them are supported out of the box and so are really needed.

They are provided to make it easier for IPS or partners to extend the datamodel with new modules and have a consistent set of XSD files. This is also important because B2B has the same set of XSD schema files and they have to match to guarantee a working process.

Please note that B2B also has the full XSD schema file set which means there is nothing to do on B2B side when adding new modules or attributes.

Informatica B2B DX receive endpoint directory

The path to the B2B DX receive endpoint directory is set by the value for "Target directory" of the "Copy export file" export post step. As this is a fixed value you should set this parameter to not editable.

Use export templates

After you have configured the export templates you can use them to create export profiles for manual export or for scheduled export jobs.

The manual export can also be triggered by an immediate export. Therefore your template must match the needs of the immediate export:

- The purpose "Available for immediate export" must be set in the template
- The export format template for the immediate export must contain the value "Editable" in the data sources for the "Catalog" and "Assortment" parameters.

Post processing

Validate

All created XML files by the export will be validated against the corresponding XSD file. The "Validate XML file(s)" export post step will cancel the export if the validation fails.

Export post step name: Validate XML file(s)

Parameter name: Cancel export in case of error

Transfer

After an export has been finished successfully, the created XML file will be transferred to the B2B DX receive endpoint directory. The path to that folder is a parameter of the "Copy export file" export post step.

Export post step name: Copy export file

Parameter name: Target directory

Export templates for IM

Data source export templates

All messages sent to the GDSN pool are transferred as XML files created by exports. For this purpose, there are four export format template files provided as mentioned below.

Catalog Request Item

The *CR_CatalogueRequest Item* export template outputs a "Catalog request" GDSN message containing items. Depending on the configuration, you can transfer new or modified trade item information with all detailed data maintained.

More information about provided data fields can be found in chapter "GDSN field list".

Allowed values: ADD, MODIFY

Note: The operation (ADD or MODIFY) specified in the GDSN message is a parameter of the export template. You have to ensure the correct value for that parameter, otherwise the pool will send an error response.

Catalog Request Link Add

The *1CR_CatalogueRequest Link ADD* export template outputs all packaging hierarchy information that has not been sent to the GDSN pool yet.

Technically, new item references of the "Next lower level" type are exported. The following rules apply:

- The referencing item (the parent item) and the referenced item (the child item) have been successfully sent at least once to the pool
- The link between the two items does not exist in the pool

Note: The export of item reference information has to analyse existing publication status data of according items. The publication status of items will be updated as a result of the response messages sent from the GDSN pool. That is why it may take a while until a new reference for a new item will be picked up by a "Link ADD" export.

Catalog Request Link Delete

The *CR_CatalogueRequest Link DELETE* export template outputs all deleted packaging hierarchy information that has been sent to the GDSN pool before.

Technically, deleted item references of the "Next lower level" type are exported. The following rules apply:

- The referencing item (the parent item) and the referenced item (the child item) have been successfully sent at least once to the pool
- The link between the two items exists in the pool
- The link between the two items has been deleted

Catalog Request Publication

The *1Sync CatalogueRequest Publication CR_CatalogueRequest Publication* export template creates a "Catalog request" GDSN message containing publications for all given items. Depending on the publication status, you can add or delete publications. Furthermore, the message can be sent for a market group or for all GLNs an item is assigned to. You can find more information in the chapter "[Publication\(see page 101\)](#)".

A publication will be sent if the items to publish follow these conditions:

- The configured target market in the export format template corresponds to the target market of the item in "Target market specific GDSN data"
- If you publish to a GLN, then the field "Trade partner" must be filled with the according customer
- If you want to publish to a market group you have to add the market group to the field "Context" and leave the field "Trade partner" blank

Note: Either the field context has to be filled with the market group or at least one "Trade partner" entry has to exist for each exported item. Otherwise an invalid XML file will be created and the export will be canceled.

As alternative you can use the *CR_CatalogueRequest Publication_ByVariables* export template export template in order to configure all the parameters manually.

Catalog Request Publication Withdrawal

The *CR_CatalogueRequest Publication HW* export template is used to withdraw a published hierarchy from the pool. It can be specified to unpublish the article from the market group or the GLN.

Afterwards the hierarchy can be changed and republished again. This has to be done with the Catalog Request Publication template.

Data recipient export templates

These export templates can only be used if you are a data recipient. To change Product 360 to the data recipient mode have a look at the chapter [Application modules](#)(see page 59).

- Catalog Item Subscription (CIS_CatalogItemSubscription.ext)
- Catalog Item Confirmation (CIC_CatalogItemConfirmation.ext)

Export template parameters

Export template parameters can be parameters of data sources, parameters of export post steps or variables. Furthermore variables can in turn be used as parameters of data sources or export post steps.

There are a lot of parameters you can configure in the export templates. All of them can be assigned to one of the following two categories:

Identify

Each of the export templates has three mandatory parameters needed for the communication with the GDSN pool. It is recommended to set the correct values and make those parameters invisible.

- "Information provider" and "Information provider GLN" respectively
This is the supplier representing your own GLN
- "Recipient" and "Recipient GLN" respectively
This is the customer representing the 1WorldSync pool GLN
- "User Id"
This is the user id to authenticate yourself against the IM pool

The "Information provider" and the "Recipient" have to exist in Product 360, otherwise the answers from the pool will not be written back to Product 360. Make sure you created the values like described in the section about the [creation of supplier and customer](#).

Select data

Most parameters are used to qualify which data should be exported.

First of all, there is a catalog and an item assortment to determine which items are to be exported.

In addition to that, the target market, a language, a price type, etc. specify which details of items are to be exported.

Export data source configuration

All GDSN export templates use the "Item list" as data source. If you want to schedule repeated export jobs, you usually want to send only new and changed items to the pool. That is why you have to change the data source of the corresponding export template.

"Changed and new items"

To change the data source to "Changed and new items" you need to do following steps:

1. Open the properties dialog of your export format template and go to the "Data sources" tab

2. Press the "New..." button and choose the "Changed and new items" data source. Then close the property dialog by pressing "OK"
3. Open the "Items" module of your export format template by double click
4. Change the data source from "Item list" to "Changed and new items" in the property view of the module
5. Go back to properties dialog and open "Data sources" tab
6. Select data source "Item list" and press "Delete" button

The following settings are recommended for the "Changed and new items" data source:

Setting	Value	Editable
Catalog	Master catalog	true
Assortment	<empty>	true
Reference date	<i>Variable "ReferenceDate", see next chapter</i>	true
Type of change	New and changed items	false
Update assortment	empty	true
GPC structure	GPC	false
Version	Working version	false

You can specify appropriate values for the visible parameters in the export profiles you create from the export templates.

It is recommended to set the "Update assortment" value to "true" for scheduled export jobs, but if you want to start a single export it may not be necessary to update the assortment.

"Items by new or changed publication status"

The export template for creating publication messages also uses the "Item list" data source by default. If you want to use the recommended publication process (see [Publication\(see page 101\)](#)), you should use another item data source for the corresponding publication export template (see [Export templates for IM\(see page 76\)](#)), the "Items by new or changed publication status" data source. The reason is that items are not changed when they are marked for publication and therefore are not picked up by the "Changed and new items" data source. The "Items by new or changed publication status" data source checks the publication status entries of items and processes all items that have new publication markings.

To change the data source to "Items by new or changed publication status" you need to do following steps:

1. Open the properties dialog of your export format template and go to the "Data sources" tab
2. Press the "New..." button and choose the "Items by new or changed publication status" data source. Then close the property dialog by pressing "OK"
3. Open the "Documents" module of your export format template by double click
4. Change the data source from "Item list" to "Items by new or changed publication status" in the property view of the module
5. Go back to properties dialog and open "Data sources" tab
6. Select data source "Item list" and press "Delete" button

The following settings are recommended for the "Items by new or changed publication status" data source:

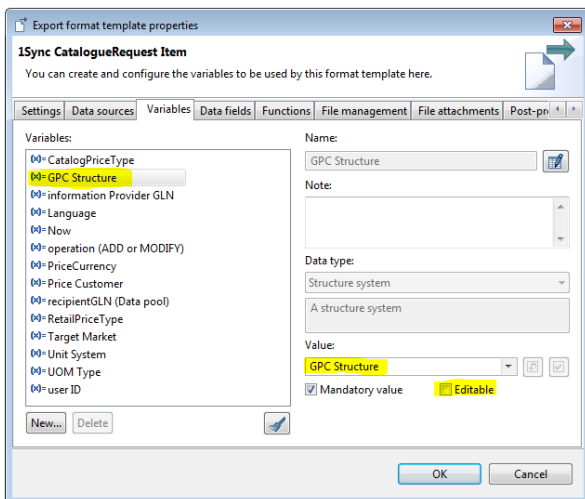
Setting	Value	Editable
Catalog	Master catalog	true
Assortment	<empty>	true
Reference date	<i>Variable "ReferenceDate", see next chapter</i>	true
Update assortment	empty	true
Version	Working version	false

You can specify appropriate values for the visible parameters in the export profiles you create from the export templates.

It is recommended to set the "Update assortment" value to "true" for scheduled export jobs, but if you want to start a single export it may not be necessary to update the assortment.

Parameters

Most parameters of the export template have to have fixed values. You should set those values to the export templates and switch the corresponding parameters to "Not editable" to hide them for further use.



If you then use an export template to create an export profile, you will see only parameters that are really needed to be configured, for example the operation "ADD" or "MODIFY" in the export profile.

Reference date

If you want a [scheduled export job](#)(see page 89) to export all items that have been created or updated after the last run, you can use a variable that gets updated with the corresponding date and time for every export job run.

First, you create a variable of type "Date and time". Then you assign this variable as value for the "Reference date" parameter of the "Changed and new items" data source. The last step is to configure the variable in an export profile: select "Update" and "after exporting" for the variable.

Export format template properties
1Sync CatalogueRequest Item

You can create and configure the variables to be used by this format template here.

Settings | Data sources | **Variables** | Data fields | Functions | File management | File attachments | Post-processing

Variables:

- CatalogPriceType
- GPC Structure
- information Provider GLN
- Language
- Now
- operation (ADD or MODIFY)
- PriceCurrency
- Price Customer
- recipientGLN (Data pool)
- ReferenceDate**
- RetailPriceType
- Target Market
- Unit System

Name: ReferenceDate

Note:

Data type: **Date and time**

A date and the time The default value is the current date. The input format is 8/18/2015 11:24 AM or 2015-08-18

Value:

☒ Mandatory value ☒ Editable

OK Cancel

Export format template properties
1Sync CatalogueRequest Item

You can select and configure the data sources to be used by this format template here.

Settings | **Data sources** | Variables | Data fields | Functions | File management | File attachments | Post-processing

Data sources:

- Changed and new items
- Item list

Name: Changed and new items

Parameter:

Catalog: Master catalog

☒ Mandatory value ☒ Editable

Assortment:

☐ Mandatory value ☐ Editable

Reference date:

ReferenceDate

New... Delete

New export profile - 1Sync CatalogueRequest Item 2

Export parameter configuration

Set the export parameters for executing this export.

Data source "Changed and new items"

Catalog:* Master catalog

Assortment:

Update assortment: Yes

Variables

Now:* Default value (8/18/2015)

ReferenceDate:* 8/18/2015 4:22 PM ☒ Update: **after exporting**

operation (ADD or MODIFY):* ADD

< Back Next > Finish Cancel

Informatica B2B DX receive endpoint directory

The path to the B2B DX receive endpoint directory is set by the value for "Target directory" of the "Copy export file" export post step. As this is a fixed value you should set this parameter to not editable.

Use export templates

After you have configured the export templates you can use them to create export profiles for manual export or for scheduled export jobs.

The manual export can also be triggered by an immediate export. Therefore your template must match the needs of the immediate export:

- The purpose "Available for immediate export" must be set in the template
- The export format template for the immediate export must contain the value "Editable" in the data sources for the "Catalog" and "Assortment" parameters

Post processing

Validate

The created XML file will be validated against the corresponding XSD file. The "Validate XML file(s)" export post step will cancel the export if the validation fails.

Export post step name: Validate XML file(s)

Parameter name: Cancel export in case of error

Transfer

After an export has been finished successfully, the created XML file will be transferred to the B2B DX receive endpoint directory. The path to that folder is a parameter of the "Copy export file" export post step.

As this is a fixed value you should set this parameter to not editable.

Export post step name: Copy export file

Parameter name: Target directory

Packaging hierarchy data providers

The pools differentiate in how they accept items and their hierarchies. The case for the DSE pool you have to send the whole hierarchy of the item on the initial export. There is no way to add the hierarchy afterward like in the IM scenario.

To be able to support this case there are 2 additional hierarchy data providers available, which can be used for all GDSN scenarios.

Complete packaging hierarchy

Note: In the GDSN scenario this data provider makes only sense when sending "Catalog Item Notification" messages to the DSE data pool.

Configuration

To configure the data provider you can look at the "Export data source configuration" on the page ["Export templates for DSE\(see page 71\)"](#).

Mode of operation

When using this data provider for an export template it will automatically export the whole hierarchy where the article is included. If the article is included in more than 1 hierarchy both hierarchies are exported.

This data provider only accepts valid hierarchies. A hierarchy is considered valid if there are no endless loops in it and if there're not more than 10 levels. (An invalid hierarchy is indicated in the packaging hierarchy view with a yellow triangle.) Otherwise this data provider will cancel the export immediately.

Data quality checks

When using a data quality run as a pre-export step, this data provider will re-evaluate the results of the data quality result.

It will only export items with the following criteria:

- The item is considered valid from DQ
- All items of the hierarchy are considered valid from DQ

All other items and hierarchies are not exported.

Hint: It is best practice to create an assortment with all items which only include the top level of the packaging hierarchies and then export the whole assortment.

Packaging hierarchy of new and changed items

Note: In the GDSN scenario this data provider makes only sense when sending "Catalog Item Notification" messages to the DSE data pool.

This data provider is very helpful when there are packaging hierarchies which are target market specific.

Configuration

To configure the data provider you can look at the "Export data source configuration" section on the chapter ["Export templates for DSE\(see page 71\)"](#).

Mode of operation

Like the "Complete packaging hierarchy" provider it checks the assortment for endless hierarchy loops and cancels the export when doing so. It also exports the whole hierarchy of the items in the assortment.

The differences to the Complete packaging hierarchy provider are the following:

- Only items within the given assortment are exported, even if a hierarchy is not complete in that assortment. That allows you to export target market specific hierarchies.
- It is possible to specify a "reference date" to the data source parameter. This allows the export to only consider hierarchies that contain items which were changed after the last export. For more information check the "Reference date" section on the chapter ["Export templates for DSE\(see page 71\)"](#).

Data quality checks

When using a data quality run as pre-export step, this data provider will re-evaluate the results of the data quality result.

It will only export items with the following criteria:

- The item is considered valid from DQ
- All items of the hierarchy are considered valid from DQ

All other items and hierarchies are not exported.

.

Technical details

Special export functions

GDSNDocumentId

The `GDSNDocumentId` and `GDSNParentDocumentId` export functions create unique identifiers for each document output in an export file. *Note:* these functions are only used and available for the IM pool.

EnumerationKey

Usually, GDSN messages contain codes for attributes that refer to a valid values list. By default, an exports output the labels for such attributes. The `EnumerationKey` export function is used to output the code instead of the label of enumeration values.

EnumerationKeyStandard

In contrast to the function `EnumerationKey`, this function can only be used for data having an enumeration assigned which is an extension of a standard enumeration. The function only returns a key if the value is contained in the standard enumeration, keys of the extended enumeration won't be returned.

This is needed for data fields that are logical keys in the data model but should be empty in the context of GDSN. Those fields got enumerations providing all valid GDSN values and a "<no code>" value.

Example

Enumeration of the "ArticlePackaging.PackagingType" field is `Enum.PackagingTypeCode.WithOptionalCode`. That enumeration is an extension of the `Enum.PackagingTypeCode` enumeration.

The function call with that field returns an empty string for the value "<No code>", and "BBG" for the field value "Bag in box".

GDSNEnumerationCode

Some GDSN attributes use other codes than the codes maintained in PIM. This function reads the GDSN codes from the corresponding configuration. It is possible to add additional codes, see [Repository configurations](#)(see page 64).

GDSNDutyFeeTaxAgencyCode

The value for `DutyFeeTaxAgencyCode` depends on the target market and the tax type. The DSE specific export function `GDSNDutyFeeTaxAgencyCode` returns the configured value for "DutyFeeTaxAgencyCode" depending on target market and tax type.

Similar to the enumeration codes you can configure the values for `DutyFeeTaxAgencyCode` in the `plugin_customization.properties` file. The pattern of such entries is

```
com.heiler.ppm.gdsn.core/GDSN.dutyFeeTaxAgencyCode.<target market code>.<tax type code> =
<DutyFeeTaxAgencyCode>
```

```
# Germany
com.heiler.ppm.gdsn.core/GDSN.dutyFeeTaxAgencyCode.276.VAT = 246
```



```
# Austria
com.heiler.ppm.gdsn.core/GDSN.dutyFeeTaxAgencyCode.040.VAT = 294
```

General approach to output data

Empty values

All optional fields don't create a XML tags in case there is no value maintained. This is needed because an error will occur for empty tags. For this the export function `CreateXMLTagWithValue` respectively `CreateXMLTagWithContent` are used.

Format numbers and dates

It is important to format the numbers and dates in the needed format of the GDSN pool.

Therefore all fields with numbers must use the `FormatDecimal` export function. This function is responsible for formatting and transferring the given number according to the specified decimal separator and the number of decimal places. The decimal separator must be always "." for decimal values. All used date fields will use the `FormatDate` export function which is formatting the given date with a given pattern to match the needed format.

Encoding

There is need for defining or prevent the encoding of strings according to their encoding setting. There are possibilities to define this actively by the export functions `IfNotEmptyThenNotEnc` or `IfEmptyThenNotEnc` and inactively by using export function which are handling the encoding already like `CreateXMLTagWithValue` which returns the transferred fragment itself as not encoded.

Conditional output

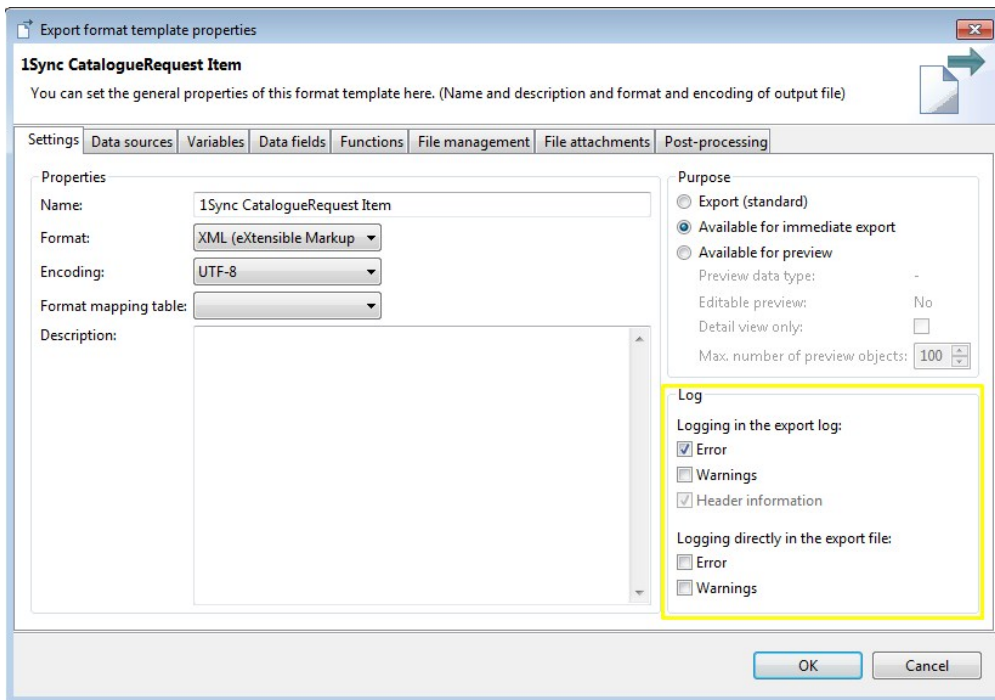
The data to output often depends on if any other specific data is set. Therefore you can define a conditional output by using export functions around the section that is to be output if the condition is fulfilled. This is possible in general by using the `IfNotEmptyThen` export function or you can use specific export functions to generate conditional output by using a Boolean comparison with the `CompareBooleanValue` export function for example. Besides this it is also possible to define a default value for some export functions.

Extend/modify export templates

Debug

There are two different possibilities to enable some debug information to be able to detect a problem.

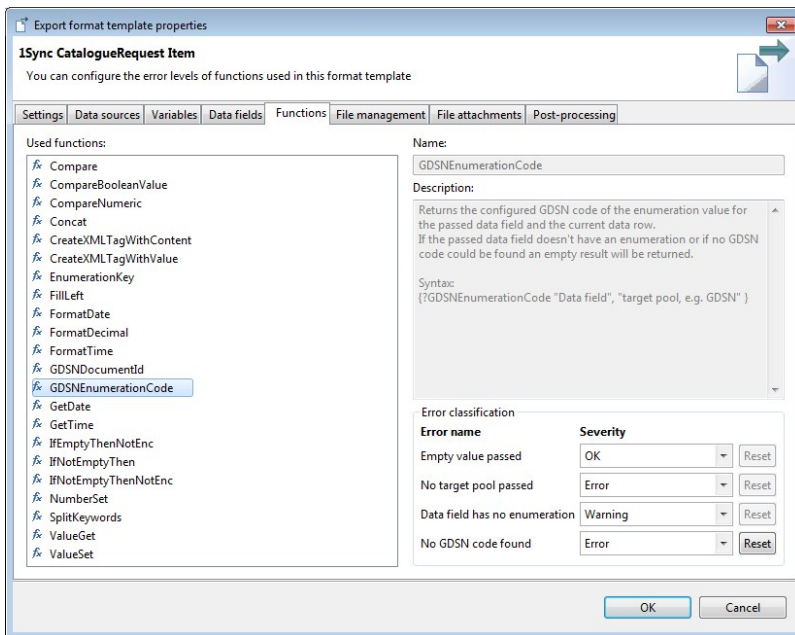
General export log



The logging in the export log defines the granularity of the logged information in the export template log which can be viewed in the process overview. By default it is only logging header information and errors. For debugging it is possible to enable the warning level as well but it is recommended to not having it active during casual business since this can change the export speed. You can also enable logging directly in the output file. You can enable the error and warning level to assist you in tracing the causes of errors during the development.

Export function

Another possibility is to change the error levels of functions. You can define the error level for some functions in the export template properties dialog which can help you to figure out a problem. Those are also helpful if you want to define a specific behavior when for example an empty value passed to the function. So this configuration can also be taken as standard modification depending on your data.



Check output

There are different ways to provide the opportunity to inspect the output XML file when errors occurs. Find two examples described in more detail below.

Add an additional "Copy export file" post-processing step before the "Validate XML file(s)" step. Define a specific target directory for this new export step. Whenever an export will be executed the export file will be copied to the target directory even if the XML validation will cause on error. But the target directory will always only contain the last executed export file.

Another way would be to disable the cancelling of the export within the "Validate XML file(s)" export step by setting the "Cancel export in case of error" to "No". Furthermore you need to change the "copy export file" target directory because you will probably send not valid XML files to the GDSN pool which will lead to errors.

With both of the mentioned examples it is possible to find out what the output is on a specific line in a XSD validation error. This makes it easier to figure out where a problem is in the data and/or template.

Besides this, it's always possible to manually transfer the export files to the pool. To put this into action the "Copy export file" export step can be deleted and after each export execution the export file can be downloaded in case of no error. This file has to be copied manually into the B2B DX receive endpoint directory.

2.1.5.2 Setup automated jobs

Usually business users don't want to care about sending their new created or modified items to the GDSN pool. Due to the fact that the Product 360 export is used to send data to the pool, you can easily define repeating export jobs which send all matching items to the pool every hour for example. The setup described below has to be done once by a skilled consultant or administrator and not touched by any business user.

Please make sure that all templates are configured already as described in the chapter [Export templates](#)(see page 69).

Which automated jobs are needed

You can create automated jobs to send all GDSN messages in a scheduled way.

Please note, that you could cause exceptions if you send data twice to the GDSN pool by manual and automated exports at the same time.

How do automated export jobs work

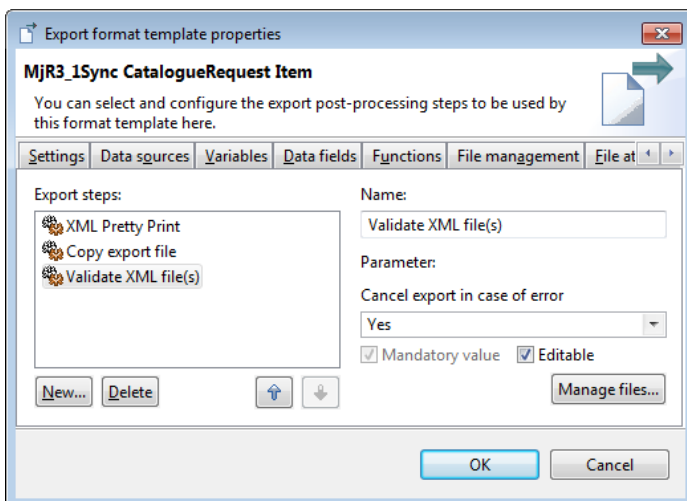
Automated jobs start exports of an export profile at specified times or in recurring intervals. The items to be sent are retrieved by the evaluation of the defined export parameters, for example assortment rules or reference date.

How to find the repeat interval for a job

There are no recommendations concerning the repeat interval of these exports. But please always have in mind that the answer from the GDSN pool could take a while and if you define the interval too short, it could be possible that a message is sent twice which will maybe result in an error.

What happens if there's no data to be exported

If an automated export job runs but there're no items to be exported, an invalid xml file will be created. To prevent such an invalid file to be transferred to B2B, the export has to be cancelled by the failing "Validate XML file(s)" export post step.



16	Mandato...	Export post-processi...	catalogueRequest	Datei "1Sync CatalogueRequest Item.xml": Fehler in Zeile
17	Summary	Post-export step	Validate XML file(s)	Export canceled due to XML validation error(s)

The next run of the job will be started as configured and pick up all items defined by the assigned assortment rules.

Automated jobs for DSE

Setting up a repeated job to automatically send all new and changed items to the GDSN pool

In the DSE GDSN scenario the item data together with their hierarchy data are sent in a single message called *CatalogItemNotification* to the GDSN pool. The corresponding export template to be used is *CIN_CatalogItemNotification*.

There are two parameters you have to specify in order to find the items to be sent. The first is the catalog and item assortment, the second is the reference date used to find all items that have been created or modified since the last execution of the job.

Note: For the initial setup you should use a date in the distant past to ensure that all item are considered. For further information see "[Export templates for DSE\(see page 71\)](#)".

Setting up a repeated job to automatically send item publications to the GDSN pool

In order to provide the item data to the customers you need to send publication messages to the GDSN pool. The first step is to mark your items for publication using the publication status as described in "[Publication\(see page 101\)](#)".

Next you have to build a dynamic assortment containing all items marked for publication, see "[GDSN Accelerator operation\(see page 94\)](#)", section "Create an assortment containing all items which are marked for publication".

Finally you have to create the export job using the export template *CIP_CatalogItemPublication* which should be already configured as described in the chapter "[Export templates\(see page 69\)](#)". Before scheduling the job set the assortment mentioned above to the "Assortment" parameter.

Automated jobs for IM

Setup repeating export job to automatically send all new items to the GDSN pool

For every repeating export job we define an export profile. The base for this export profile is an assortment containing all items which are not sent to the GDSN pool yet. The chapter "[GDSN Accelerator operation\(see page 94\)](#)" describes how to create this kind of assortment. After you have created the assortment open the "Explorer (items)" perspective, go the "Context selection" tab, right-click on "Export profiles" and select "New export profile...". The dialog in the screenshot below pops up where you are able to define a name and select a format template.

New export profile - 1Sync CatalogueRequest Item 2

Create new export profile
You can create a new export profile here. First of all, select a format template.

Format template: 1Sync CatalogueRequest Item

Profile name: 1Sync CatalogueRequest Item ADD

Profile description: This export profile is based on the format template "1Sync CatalogueRequest Item".
It is intended to send all NEW items to GDSN data pool

< Back Next > Finish Cancel

The chosen export format template should be already configured as described in the chapter "[Export templates\(see page 69\)](#)" and so there should be only a few configurations left. You can choose the according catalog but it is highly recommended that you only send items from your master catalog. Additionally you can choose an assortment, which should be of course the one you just created in the step before. Last but not least the operation has to be defined, which will be set to ADD in this case because we want to "add" new items to the GDSN pool.

Edit export profile - 1Sync CatalogueRequest Item

Export parameter configuration
Set the export parameters for executing this export.

Data source "Changed and new items"

Catalog:* Master catalog

Assortment: GDSN - Items which are NOT in the GDSN p

Variables

operation (ADD or MODIFY):* ADD

< Back Next > Finish Cancel

After you have successfully created an export profile you can schedule a repeating export job. To do that, right-click on the created export profile and select "Export". The parameters should already be well defined so you only have to configure the interval options on the last wizard page. An example is shown below:

Export export profile - 1Sync CatalogueRequest Item

Select output destination
Plan the execution time of the export here and specify additional settings based on this.

Execution: Schedule export (repeating) From 8/18/2015 at 11:08 AM every 1 Hours.

Start date: 8/18/2015

Start time: 11:08 AM

Comment:

☐ Save export file(s) locally

Which files do you want to download?

☐ Data file(s) (1 files)

☐ Multimedia archive

Do you want the data files to be packed in a ZIP archive?

☐ Pack file(s)

Which local directory do you want to download the files to?

Directory:

< Back Next > Finish Cancel

Setup repeating export job to automatically send all changed items to the GDSN pool

This scenario is nearly the same as before with the small difference that the item is edited instead of created. One main difference is the assortment which should contain of course all items which are already in the GDSN pool. Please see chapter "[GDSN Accelerator operation\(see page 94\)](#)" and create this assortment. After this, create a new export profile but compared to the new item scenario use your new assortment and set the operation to MODIFY. It is also recommended to name the export profile clearly (e.g. "CR_CatalogueRequest Item MODIFY").

Export export profile - 1Sync CatalogueRequest Item MODIFY

Export parameter configuration
Set the export parameters for executing this export.

Data source "Changed and new items"

Catalog:* Master catalog

Assortment: GDSN - Items which are already in the GDSN

Variables

operation (ADD or MODIFY):* MODIFY

< Back Next > Finish Cancel

Last but not least schedule a repeating export job based on the created export profile and define a corresponding interval.

Please note: When you add or delete a reference and have a job which sends updates automatically, a "Link add" or "Link delete" will be sent. Due to the fact that the item has changed, also a "Item modify" message will be sent to the GDSN pool. This is unnecessary and a technical limitation but does not affect your data in the system or in the GDSN pool.

Setup repeating export job to automatically send all created packaging hierarchies to the GDSN pool

In the 1WorldSync US scenario it is required to send "Link" messages to create or delete packaging hierarchies. As pre requirement all items which should be linked as a hierarchy have to be added to the GDSN pool first. To ensure this, it is recommended that you use an assortment which contains only items which are already in the GDSN pool (see chapter "[GDSN Accelerator operation\(see page 94\)](#)", if you already created this assortment you can also reuse it).

In a next step you have to create an export profile as in all other scenarios. This time you need a different export template named *CR_CatalogueRequest Link ADD* which should be already configured as described in the chapter "[Export templates\(see page 69\)](#)". Furthermore select the catalog and the corresponding assortment

and create afterwards a repeating export job (please find more detailed information in the first chapter as there is a more detailed description).

Please note:

- There is no "Link modify" scenario what means that if you want to change any information related to a reference you first have to send a "Link delete" message followed by a new "Link add" message with the new values.
- The "Link" messages are one of the most important differences between the 1WorldSync and the GDSN scenario. There are no link messages in the GDSN scenario because the hierarchies are already defined in the CIN messages.

Finding the repeat interval for sending all created packaging hierarchies

Due to the fact that the "Link" messages are only sent for items which are already in the pool, it makes sense to schedule the jobs with a time lag to the job which sends new created items to the GDSN pool. As it takes the longest time to add newly created items to the GDSN pool it is recommended that you schedule the "Link add" job after the "Item add" job with an interval according to your data volume. That means if you are sending thousands of new items to the GDSN pool you probably have to define a longer interval than if you only send ten items.

Setup repeating export job to automatically send all deleted packaging hierarchies to the GDSN pool

If you remove an item from the packaging hierarchy a "Link delete" message should be sent to update the changes in the GDSN pool. The scenario is basically the same as the "Link add" message but we have a different export format template. First create (or reuse) the assortment as described in the chapter "[GDSN Accelerator operation\(see page 94\)](#)" / "Create an assortment containing all items which have been sent to the pool". After you have created the assortment use it within a new export profile. In addition select the export template *CR_CatalogueRequest Link DELETE*. As in all other scenarios create a repeating export job based on this profile and define a corresponding interval.

Note: It probably makes sense not to schedule the job the same time you send all created packaging hierarchies to not get confused but there is no real need to do that.

Setup repeating export job to automatically send item publications to the GDSN pool

In order to provide the item data to the customers you need to send publication messages to the GDSN pool. The first step is to mark your items for publication using the publication status as described in "[Publication\(see page 101\)](#)".

Next you have to build a dynamic assortment containing all items marked for publication, see "[GDSN Accelerator operation\(see page 94\)](#)".

Finally you have to create an export profile as in all other scenarios. This time you need a different export template named *CR_CatalogueRequest Publication* which should be already configured as described in the chapter "[Export templates\(see page 69\)](#)". Furthermore select the catalog and the corresponding assortment and create afterwards a repeating export job.

2.1.6 GDSN Accelerator operation

2.1.6.1 Maintain valid data

Item identifiers: GTINs

GTINs for GDSN and food and beverage data

All items which should be sent to the pool **must** have 14 digits GTINs. Otherwise there will be no answers from the pool written back to Product 360.

When the GTIN of an item which should be sent does not contain 14 digits you can add leading zeros until it matches the 14 digits criteria.

Maintain data

Product 360 contains several perspectives to maintain *GDSN* and *Food and beverage* data. These are described below.

GDSN (core)

In the *GDSN (core)* perspective you find the views to maintain the most important aspects of GDSN. These include the target market specific data, the item references to build the packaging hierarchy and the *Packaging hierarchy* view to see a visualization of the packaging hierarchy, *Trade item lifespan information* and the *Publication status* view to see in which state your item is regarding the GDSN message chain. *Sustainability information* view is only available for the DSE pool.

GDSN (supply chain data)

This perspective enables you to maintain all data that is relevant in the supply chain. Views to maintain data of the following modules are included: Packaging-specific data, Trade item handling, Data carrier information, Delivery purchasing information

GDSN (Canada-specific data)

This perspective is only available for the IM pool. It contains the same views as the *GDSN (core)* perspective plus two additional views to maintain the data of the Canada extension.

Food and beverage (core)

In this perspective you can maintain the most important data of your food or beverage. These include diet related information, allergen related information, nutrient information and ingredients.

Food and beverage (misc)

This perspective contains views to maintain data about special aspects of the product. For example preparation and serving related information but also physiochemical and microbiological information. Additionally there are views to maintain data about certifications you might have.

Ingredients

You can maintain the ingredients of your product in the *Food and beverage (core)* perspective, however, first you have to make them available for the whole system in this perspective. You can think of this perspective as the pool of ingredients you use and the data of your product only references ingredients in this pool.

Certifications

This is the pool of certifications you can use in the system. You can add references to these certifications in two places. First, you can add a certification to your product in the *Item certification* view and second you can add a certification for a specific diet type in the *Diet related information* view.

Third parties

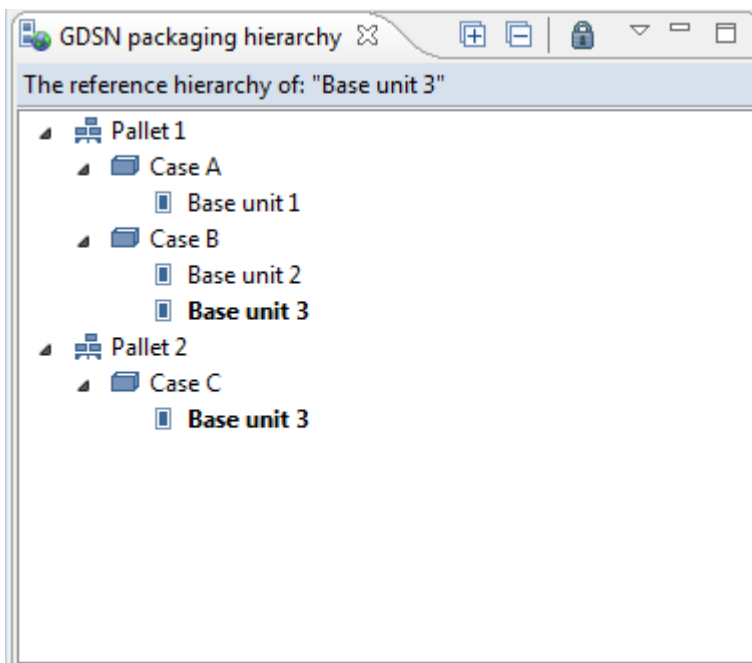
Third parties are all entities you interact with which are neither customers nor suppliers. There are three categories of third parties in the standard, Brand owner, Certification organisation and Manufacturer of goods. For use with certifications, it is necessary to add the category Certification organisation, otherwise they won't be shown when you create a certification.

Valid values

If there are valid value lists given by GDSN and/or the respective pool, only those values will be shown in the drop down lists of the field. In case Product 360 requires a value for the specific field but GDSN does not, there is an additional value <No code> contained in the proposal list.

The packaging hierarchy view

The "GDSN packaging hierarchy" view can be used to visualize the packaging hierarchies of items:



The view shows the complete packaging hierarchy outgoing from one or more items. The hierarchy includes all higher level and all lower level items of each outgoing item. The outgoing items are highlighted in the view. The content of the view depends on the selection. This means that each time one or more items are selected in any other view the packaging hierarchy view loads its content based on the selected items. The view can be locked to retain the current content. The functionality "Obtain content exclusively from" is

available in the view to link it with a specific view. The labels of the tree nodes which represent the items of the hierarchy can be configured by using the view menu entry "Configure node labels...". Up to five fields can be defined to be used as part of the node labels. By a double-click on an item in the view the current outgoing item of the displayed hierarchy can be changed. In this way the user can "navigate" through the complete packaging hierarchy and see possible other hierarchies for items which are contained in more than one packaging hierarchy. The packaging hierarchy view allows multi-selection which can be used e.g. to show all items of the current hierarchy in a separate item view (by using the context menu entry "Show selected items"). It can be used e.g. for maintenance of all items of a packaging hierarchy at once, since the packaging hierarchy view itself is read-only.

Validate data

There are three levels of data validations for *GDSN* and *Food and beverage* data in Product 360:

Data model validation

The Product 360 data model and repository configuration ensures for example the correct field length and valid value lists.

Data quality checks

There is a whole package checking rules given by GDSN or IM. Most of them check dependencies between two or more field values.

Examples:

- Check 'Canceled date' is empty if 'Discontinued date' is populated
- Check 'Is dispatch unit' items have a 'Gross weight (imperial)' populated
- Check 'Number of items per layer' is greater than zero if 'Number of layers per pallet' is greater than zero

XSD validation

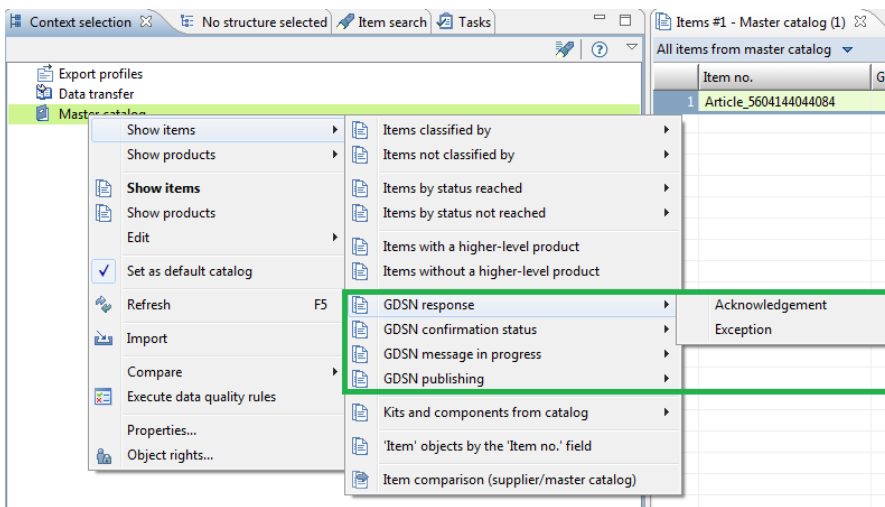
The XSD validation is executed during the export of the data. It is a final check, that the output has the correct format.

Example: Check if mandatory fields have values.

2.1.6.2 Send data

Use queries to get an overview of your data

Queries are provided to find items based on their status related to the GDSN pool (using the publication status). That means the user has the possibility to easily find all items which have already been sent to the pool successfully or those where the GDSN pool has sent an exception for. To show the different kinds of possible queries just right click on a catalog and select "Show items" and see all submenu entries starting with "GDSN".



The queries are separated into four categories:

- **GDSN response:** Find all items which got a specified response from the GDSN pool, e.g. all items which received an "Acknowledgement" response for an "Item add" message or all items which received an "Exception" response for a "Link delete" message.
- **GDSN confirmation status:** These queries find all items with a specific severity of item authorization response messages. It is used as a helper to be able to keep an overview over specific responses of the data recipients.
- **GDSN message in progress:** Find all items which are currently processed, that means all items which have been sent to the pool with any message, e.g. "Link add" but not received a response yet. Due to the fact that it can take a while until a response from the GDSN pool is received these queries are useful to determine if the item has already been sent. This query category is only available for the IM pool.
- **GDSN publishing:** Find all items which should be published, republished or unpublished.

Build dynamic assortments from queries

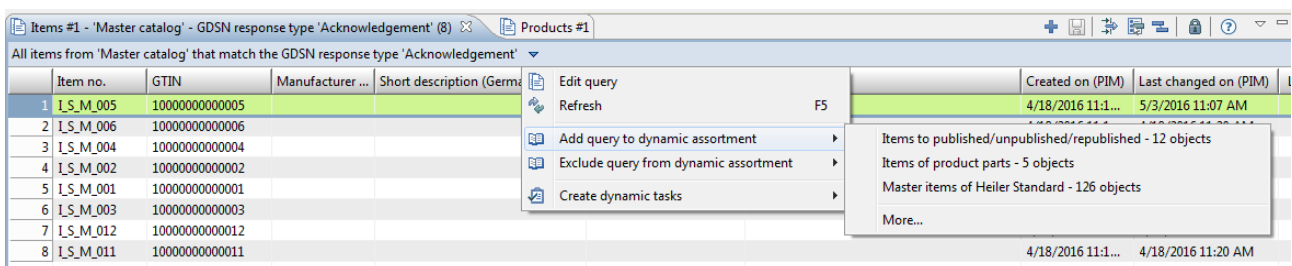
All items which have been added to the pool

Step 1: Find all items that have been added to the pool

Choose "GDSN response - Acknowledgement" query from the context menu. In the query configuration dialog set "Message type" to "Item response" and then execute the query.

Step 2: Create an assortment from the result set

Select "Add query to dynamic assortment" from table header context bar and create new assortment using "More...".



Step 3: Find all items which have been added to the pool and got a modify exception

Choose "GDSN response - Exception" query from the context menu. In the query configuration dialog set "Message type" to "Item response", "Executed operation" to "Modify" and then execute the query.

Step 4: Add items to the assortment

Select "Add query to dynamic assortment" from table header context bar and choose your new assortment.

If everything was configured correctly you can verify the assortment by clicking on "View->Item assortments". Your assortment should look similar to the picture below:

Items which are in GDSN pool	82 items
Items from 'Master catalog' with GDSN response type 'Acknowledgement', 'Item response'	78 items
Items from 'Master catalog' with GDSN response type 'Exception', 'Item response', 'Modify'	4 items

All items which have not been added to the pool

Step 1: Create an item assortment containing all items that are to be transferred to the GDSN pool

This could be all items that are mapped to the GPC structure system

Step 2: Find all items that have been added to the pool

See above

Step 3: Exclude all items that have been added to the pool from the assortment

Select "Exclude query from dynamic assortment" from table header context bar and choose your new assortment.

The screenshot shows a table titled "All items from 'Master catalog' that match the GDSN response type 'Acknowledgement'". The table has columns: Item no., GTIN, Manufacturer..., and Short description (German). A context menu is open over the table, showing options: Edit query, Refresh (F5), Add query to dynamic assortment, Exclude query from dynamic assortment, and Create dynamic tasks. The "Exclude query from dynamic assortment" option is selected, and a sub-menu is visible showing: Items to published/unpublished/republished - 12 objects, Items of product parts - 5 objects, Master items of Heiler Standard - 126 objects, and More...

Step 4: Find all items which have been added to the pool and got a modify exception

See above

Step 5: Exclude items from the assortment

See above

All items which are marked for publication

To build an assortment with all items which are marked for publication you have to use the "GDSN publishing" query three times and add each result to an assortment

Step 1: Find all items that are marked to be published

Choose "GDSN publishing - To be published" query from the context menu and execute the query.

Add the result to a new assortment using "Add query to dynamic assortment" from table header context bar.

Step 2: Find all items that are marked to be unpublished

Choose "GDSN publishing - To be unpublished" query from the context menu and execute the query.

Add the result to your new created assortment using "Add query to dynamic assortment" from table header context bar.

Step 3: Find all items that are marked to be republished

Execute step 2 with the "GDSN publishing - To be republished" query.

If everything was configured correctly you can verify the assortment by clicking on "View->Item assortments". Your assortment should look similar to the picture below:

Items to published/unpublished/republished	12 items
<input checked="" type="checkbox"/> Items from 'Master catalog' with GDSN message type 'To be published'	12 items
<input checked="" type="checkbox"/> Items from 'Master catalog' with GDSN message type 'To be republished'	0 items
<input checked="" type="checkbox"/> Items from 'Master catalog' with GDSN message type 'To be unpublished'	0 items

Send item data

In GDSN you first ADD, LINK and MODIFY your data in the pool before you PUBLISH it to your customers. For all these steps you create a message to the pool using the Product 360 export with the corresponding export templates described in ["Export templates"](#) (see page 69). The export works on an assortment of items. To build an assortment you can, for example, use the search. Further information on assortment creation can be found in the Product 360 user manual.

You can either execute each of the export templates manually or automated by using a scheduled job. How to do that is described in ["Setup automated jobs"](#) (see page 87).

Publish data

You most likely want to publish your data in an automated way using a job. To make that work, you have to mark the items which have to be published, so that the job can identify them. The items can be marked by importing a specific publication status as described in ["Publication"](#) (see page 101).

2.1.6.3 Receive feedback

With the publication status you are able to see the current status in the synchronization process of each item. This means you are able to indicate if the item is in the GDSN pool, to which customer the item is published or if the item has already been transferred to the GDSN pool but no response has arrived yet. Furthermore it is easily possible to find out problems for a specific item or regarding the hierarchy. Therefore there are different "Message types" which indicate the current state of the item, for example

- Item response
 - Item was received by the GDSN pool which validated the data and answered with this status information
- Publication response
 - The publication for this item was delivered to the GDSN pool which answered with this status information
- CIC response

- Feedback message from the target market or customer for an item which you published. The CIC responses are different for DSE and IM.

- **DSE**

For this message there will be two publication status entries in Product 360:

1. A status containing a context: This publication status includes publication messages with errors which are item specific. The context indicates the item with the highest hierarchy level.
2. A status without context: This publication status informs the user about the general status of the CIC and is only present in the item with the highest hierarchy level. No publication status messages are shown.



With B2B GDSN Accelerator 10.1.0.1 the mapping for the "Trade partner" got changed. From now on the recipient of the data will be mapped instead of the customer with the pool GLN. This change is leading to clear information about the current state of the hierarchy for each customer.

- **IM**

The CIC publication status is only written in the item with the highest hierarchy level and contains all publication messages for the items in his hierarchy.

When the pool responds with an exception there could be different reason for that. To see all relevant exceptions for an item you can double click on the message in the publication message view to see all messages which were sent from the pool.

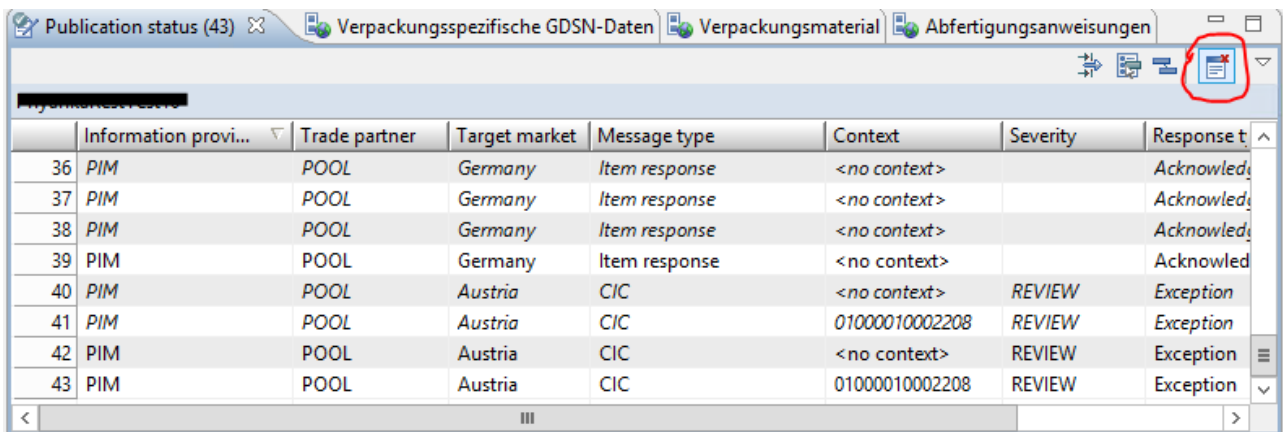
The screenshot shows the Informatica MDM interface. A 'Message' dialog box is open, displaying a list of error messages. Below the dialog, the 'Publication status' table is visible, showing the status of the item.

Information provi...	Trade partner	Target market	Message type	Context	Severity	Response type
1	PIM	POOL	Worldwide	Item response	<no context>	Acknowledger
2	PIM	POOL	Germany	Item response	<no context>	Acknowledger
3	PIM	POOL	Austria	CIC	<no context>	Exception
4	PIM	POOL	Austria	CIC	01000010002208	Exception



Publication status entries

Only the newest entries are shown in the publication status view for an item. If you want to see all messages including the ones which were overwritten you can click on the button marked in **red** and see the old entries written in *italic*.



	Information provi...	Trade partner	Target market	Message type	Context	Severity	Response t
36	PIM	POOL	Germany	Item response	<no context>		Acknowledg
37	PIM	POOL	Germany	Item response	<no context>		Acknowledg
38	PIM	POOL	Germany	Item response	<no context>		Acknowledg
39	PIM	POOL	Germany	Item response	<no context>		Acknowledg
40	PIM	POOL	Austria	CIC	<no context>	REVIEW	Exception
41	PIM	POOL	Austria	CIC	01000010002208	REVIEW	Exception
42	PIM	POOL	Austria	CIC	<no context>	REVIEW	Exception
43	PIM	POOL	Austria	CIC	01000010002208	REVIEW	Exception

2.1.6.4 Publication

After item data has been added to the GDSN pool and the hierarchies were created, it is possible to publish it for a specific customer (data recipient), market groups (IM only) or target markets (DSE only) by sending a GDSN publication message. By doing this the data recipient has the ability to view and synchronize the published item including all the child items below that item. An existing publication for a specific customer, market group or target market can be deleted.

Note: Only the root item of a hierarchy has to be published actively. The rest of the hierarchy will be published with it automatically. The root item of a hierarchy may be target market specific.

After an item has been published and subscribed to by a data recipient, some additional validations are triggered for the published GTINs and their hierarchies. If an item or a hierarchy fails due to incorrect or insufficient data, a corresponding response GDSN message will be generated by the GDSN pool. The Informatica B2B DX proceeds this file and creates a publication status entry for the affected item.

In order to make the publication process more traceable, you shouldn't publish individual items manually but use the approach described below.

Mark items for publication

After the item data and item hierarchies are maintained completely and have been sent to the GDSN pool they are ready for publication. Some items may have to be unpublished since they are deprecated, some may have to be republished to a certain data recipient.

You can add all that information to the affected items using an import, it is not possible at the moment to add those information in the Product 360 UI directly. The detailed description of the data to be imported is described in the pool specific chapters below.

Publish items automatically

Once you've added the publication status information to the items, the next run of the scheduled publication export job picks up all that data and creates corresponding GDSN messages. Of course, it's also possible to start a publication export manually.

Note: GDSN publication messages are only created for items that had been sent to the GDSN pool before. Even if an item is marked for publication, it won't be published if it hasn't been sent to the GDSN pool successfully.

There's a special algorithm which analyses the publication status entries created by the import of publication markings and by responses sent by the GDSN pool. The following rules apply:

- In general, a publication message will only be created if the item has successfully been sent to the GDSN pool before.
- If there's a publication status of message type "Item response" or "Publication response" created after the publication marking, no publication message will be created
- Publication messages of type "republish" or "unpublish" will only be created if the item has been published successfully before

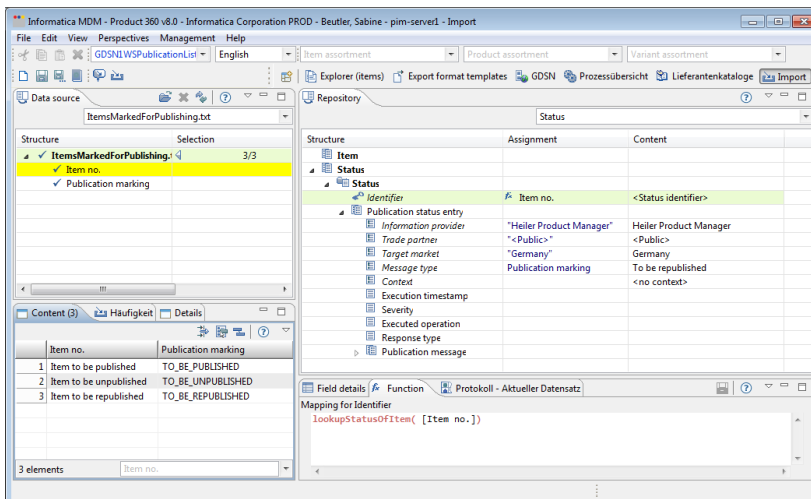
Publication process for DSE

Mark items for publication (DSE)

You can publish items for a specific customer (data recipient) or for target markets. If you want to publish items for a target market you have to choose the "<Public>" customer as trade partner, otherwise select the desired customer.

Note: All customers maintained in your Product 360 should have a valid GLN, it's needed to identify them as trade partner against the GDSN pool.

All you need for the import of publication markings for items is a file containing the item numbers or GTINs and the type of publication operation you want to create (publish, unpublish, republish).



Note: Publication status information objects are independent objects and mapped to items. That's why you need a special import function to find the corresponding status entries for the items.

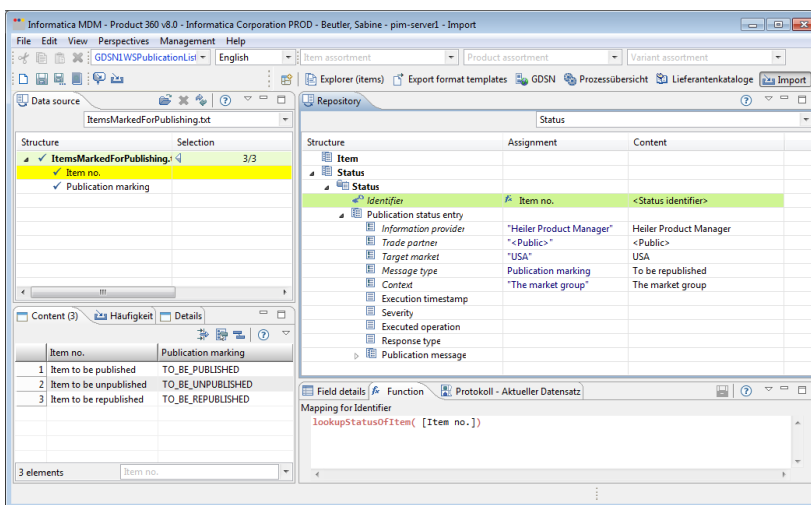
Publication process for IM

Mark items for publication (IM)

You can publish items for a specific customer (data recipient) or for a market group. If you want to publish items for a market group you have to map the name of the market group to the *Context* field, otherwise select the desired customer.

Note: All customers maintained in your Product 360 should have a valid GLN, it's needed to identify them as trade partner against the GDSN pool.

All you need for the import of publication markings for items is a file containing the item numbers or GTINs and the type of publication operation you want to create (publish, unpublish, republish).



Note: Publication status information objects are independent objects and mapped to items. That's why you need a special import function to find the corresponding status entries for the items.

2.1.7 GDSN Accelerator field list

GDSN_MjR3_FieldList.xlsx

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleMicrobiologics.OrganismCode	Organism code	Organismus-Code	Microbiology	Mikrobiologie Angaben	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeveragePropertiesInformationModule/microbiologicalInformation/microbiologicalOrganismCode	FAB
ArticleMicrobiologicsUOM.Unit	Unit	Maßeinheit	UOM	Maßeinheiten		FAB
ArticleMicrobiologicsUOM.OrganismMaximumValue	Organism maximum value ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Organismus Maximalwert ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Maßeinheiten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeveragePropertiesInformationModule/microbiologicalInformation/microbiologicalOrganismMaximumValue	FAB
ArticleMicrobiologicsUOM.OrganismReferenceValue	Organism reference value ({ArticleDomainUOMType.LK.UOMType#	Organismus Referenzwert ({ArticleDomainUOMType.LK.UOMType#	UOM	Maßeinheiten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeveragePropertiesInformationModule/	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	Type.LK.UOMType#UOM type selectable}	Einheitentyp wählbar}}			microbiologicalInformation/microbiologicalOrganismReferenceValue	
ArticleMicrobiologicsUOM.OrganismWarningValue	Organism warning value ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Organismus Warnwert ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Maßeinheiten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeveragePropertiesInformationModule/microbiologicalInformation/microbiologicalOrganismWarningValue	FAB
ArticleDietLang.DietTypeDescription	Description ({ArticleDomainLangType.LK.Language#Language selectable})	Beschreibung ({ArticleDomainLangType.LK.Language#Sprache wählbar}}	Language specific data	Sprachspezifische Daten	dietInformationModule/dietInformation/dietTypeDescription	FAB
ArticleDietList.DietTypeCode	Diet type	Ernährungsart	Diets	Ernährungsarten	dietInformationModule/dietInformation/dietTypeCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleDietList. DietSubtypeCode	Diet subtype ({ArticleSubDomainType.LK.Std_LK_Text250_01#Diet type code selectable})	Ernährungsunterart ({ArticleSubDomainType.LK.Std_LK_Text250_01#Ernährungsart wählbar})	Diets	Ernährungsarten	dietInformationModule/ dietInformation/dietTypeInformation/ dietTypeSubcode	FAB
ArticleDietList. Certification	Certification ({ArticleSubDomainType.LK.Std_LK_Text250_01#Diet type code selectable})	Zertifizierung ({ArticleSubDomainType.LK.Std_LK_Text250_01#Ernährungsart wählbar})	Diets	Ernährungsarten	Reference to certification entity with these fields: dietInformationModule/ dietInformation/dietTypeInformation/ dietCertification/ certificationOrganisationIdentifier, dietInformationModule/ dietInformation/dietTypeInformation/ dietCertification/ certificationAgencydietInformationModule/ dietInformation/ dietTypeInformation/ dietCertification/ certificationStandard,	FAB
ArticleAllergen. AllergenSpecificationAgency	Allergen specification agency	Allergenspezifizierungsagentur	Allergenrelated information	Allergenbezogene Informationen	allergenInformationModule/ allergenRelatedInformation/ allergenSpecificationAgency	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			mat ion			
ArticleAllergen.AllergenSpecificationName	Allergen specification name	Allergenspezifikationsname	Allergenbeziehungsinformation	Allergenbeziehungsinformationen	allergenInformationModule/allergenRelatedInformation/allergenSpecificationName	FAB
ArticleAllergenLang.AllergenStatement	Allergen statement ({ArticleDomainLangType.LK.Language#Language#Language#selectable})	Allergenhinweise ({ArticleDomainLangType.LK.Language#Sprache#wählbar})	Language specification	Sprachspezifische Daten	allergenInformationModule/allergenRelatedInformation/allergenStatement	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ata			
ArticleAllergenList.AllergenTypeCode	Allergen type	Allergenart	Allergens	Allergene	CatalogueItemNotification/ CatalogueItem/tradeItem/ tradeItemComponents/ allergenInformationModule/ allergenRelatedInformation/allergen/ allergenTypeCode	F A B
ArticleAllergenList.LevelOfContainmentCode	Level of containment ({ArticleSubDomainType.LK.Std_LK_Text250_01#Allergen type selectable})	Grad des Vorkommens ({ArticleSubDomainType.LK.Std_LK_Text250_01#Allergenart wählbar})	Allergens	Allergene	CatalogueItemNotification/ CatalogueItem/tradeItem/ tradeItemComponents/ allergenInformationModule/ allergenRelatedInformation/allergen/ LevelOfContainmentCode	F A B
ArticlePreparationServing.PreparationType	Preparation type	Zubereitungsart	Preparationsservicing information for	Zubereitung	CatalogueItemNotification/ CatalogueItem/tradeItem/ tradeItemComponents/ foodAndBeveragePreparationServingModule/preparationServing/ preparationTypeCode	F A B

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			mat ion			
ArticlePreparationServing.ConvenienceLevel	Convenience level	Conveniencegrad	Preparation serving information	Zubereitung	foodAndBeveragePreparationServingModule/preparationServing/convenienceLevelPercent	FAB
ArticlePreparationServingUOM.maximumOptimumConsumptionTemperature	Maximum optimum consumption temperature (ArticleDomainUOMType.LK.UOMType#UOM type	Verzehrtemperatur (Max.) (ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar))	Preparation serving	Maßeinheiten	foodAndBeveragePreparationServingModule/preparationServing/maximumOptimumConsumptionTemperature	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	selectable})		information UOM			
ArticlePreparationServingUOM.maximumOptimumConsumptionTemperatureUOM	Maximum optimum consumption temperature UOM ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Verzehrtemperatur (Max.): Maßeinheit ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Preparation serving information UOM	Maßeinheiten		FAB
ArticlePreparationServingUOM.minimumOp	Minimum optimum consumpti	Verzehrtemperatur (Min.) ({ArticleDoma	Prep	Maßeinheiten	foodAndBeveragePreparationServingModule/preparationServing/	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
minimumConsumptionTemperature	on temperature (ArticleDomainUOMType.LK.UOMType#UOM type selectable)	inUOMType.LK.UOMType#Einheitentyp selectable)	ratio serving information UOM		minimumOptimumConsumptionTemperature	
ArticlePreparationServingUOM.minimumOptimumConsumptionTemperatureUOM	Minimum optimum consumption temperature UOM (ArticleDomainUOMType.LK.UOMType#UOM type selectable)	Verzehrtemperatur (Min.): Maßeinheit (ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar)	Preparation ratio serving information	Maßeinheiten		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			at ion U O M			
ArticlePreparationServingLang.Precautions	Precautions (<code>{ArticleDomainLangType.LK.Language#Language selectable}</code>)	Vorsichtsmaßnahmen (<code>{ArticleDomainLangType.LK.Language#Sprache wählbar}</code>)	L a n g u a g e- s p e c i f i c d a	Sprachspezifische Daten	foodAndBeveragePreparationServingModule/preparationServing/preparationConsumptionPrecautions	F A B
ArticlePreparationServingLang.PreparationInstructions	Preparation instructions (<code>{ArticleDomainLangType.LK.Language#Language selectable}</code>)	Zubereitungsanweisungen (<code>{ArticleDomainLangType.LK.Language#Sprache wählbar}</code>)	L a n g u a g e- s p e c i f i c d	Sprachspezifische Daten	foodAndBeveragePreparationServingModule/preparationServing/preparationInstructions	F A B

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ata			
ArticlePreparationServingLang.ServingSuggestion	Serving suggestion ({ArticleDomainLangType.LK.Language#Language#Language#selectable})	Servivorschlag ({ArticleDomainLangType.LK.Language#Sprache#wählbar})	Language specification data	Sprachspezifische Daten	foodAndBeveragePreparationServingModule/preparationServing/servingSuggestion	FAB
ArticlePreparationServingProductYield.ProductYieldType	Product yield type	Art der Ergiebigkeit	Product yield information	Ergiebigkeit des Produktes	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeveragePreparationServingModule/preparationServing/productYieldInformation/productYieldTypeCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticlePreparationServingProductYield.ProductYieldVariationPercentage	Product yield variation [%]	Ergiebigkeit: Abweichung [%]	Product yield information	Ergiebigkeit des Produktes	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeveragePreparationServingModule/preparationServing/productYieldInformation/productYieldVariationPercentage	FAB
ArticlePreparationServingProductYieldUOM.ProductYield	Product yield ({ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Ergiebigkeit: Wert ({ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Product yield UOM	Maßeinheiten	foodAndBeveragePreparationServingModule/preparationServing/productYieldInformation/productYield	FAB
ArticlePreparationServingProductYieldUOM.ProductYieldUOM	Product yield UOM ({ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Ergiebigkeit: Maßeinheit ({ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Product yield	Maßeinheiten		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	selectable})		d U O M			
ArticleNutrient.PreparationState	Preparation state	Zubereitungsgrad	Nutritional information	Nährwertangaben	nutritionalInformationModule/nutritionHeader/preparationStateCode	FAB
ArticleNutrient.Lang.ServingSizeDescription	Serving size description ({ArticleDomainLangType.LK.Language#Language#Language#selectable})	Portionsgröße: Beschreibung ({ArticleDomainLangType.LK.Language#Sprache#wählbar})	Language specific data	Sprachspezifische Daten	nutritionalInformationModule/nutritionHeader/servingSizeDescription	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleNutrientLang.DailyValueIntakeReference	Daily value intake reference ({ArticleDomainLangType.LK.Language#Language#Language selectable})	Tagesdosis Referenzwert ({ArticleDomainLangType.LK.Language#Sprache wählbar})	Language specific data	Sprachspezifische Daten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/nutritionalInformationModule/nutritionHeader/dailyValueIntakeReference	FAB
ArticleNutrientBasisQuantity.NutrientBasisQuantity	Nutrient basis quantity ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Bezugsgröße ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Nutrient basis quantity	Basisnährstoffmenge	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/nutritionalInformationModule/nutritionHeader/nutrientBasisQuantity	FAB
ArticleNutrientBasisQuantity.NutrientBasisQuantityUOM	Nutrient basis quantity UOM	Bezugsgröße: Maßeinheit ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Nutrient basis quantity	Basisnährstoffmenge		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	{{ArticleDomainUOMType.LK.UOMType#UOM type selectable}}	K.UOMType# Einheitentyp wählbar}}	nt basis quantity			
ArticleNutrientBasisQuantity.ServingSize	Serving size {{ArticleDomainUOMType.LK.UOMType#UOM type selectable}}	Portionsgröße {{ArticleDomainUOMType.LK.UOMType# Einheitentyp wählbar}}	Nutrient basis quantity	Basisnährstoffmenge	CatalogueItemNotification/ CatalogueItem/tradeItem/ tradeItemComponents/ nutritionalInformationModule/ nutritionHeader/servingSize	FAB
ArticleNutrientBasisQuantity.ServingSizeUOM	Serving size UOM {{ArticleDomainUOMType.LK.UOMType#UOM type selectable}}	Portionsgröße: Maßeinheit {{ArticleDomainUOMType.LK.UOMType# Einheitentyp wählbar}}	Nutrient basis	Basisnährstoffmenge		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			quantity			
ArticleNutrientList.NutrientTypeCode	Nutrient type	Nährstoffart	Nutrients	Nährstoffe	nutritionalInformationModule/nutritionHeader/nutrientDetail/nutrientTypeCode	FAB
ArticleNutrientListQuantity.NutrientBasisQuantityType	Nutrient basis quantity type	Art der Bezugsgröße	Nutrient list quantity	Messwerte der Nährstoffe	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/nutritionalInformationModule/nutritionHeader/nutrientBasisQuantityTypeCode	FAB
ArticleNutrientListQuantity.QuantityContained	Quantity contained ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nutrient type selectable},	Enthaltene Menge ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nährstoffart wählbar}, {ArticleSubDomainUOMType.LK.Std_LK_	Nutrient list quantity	Messwerte der Nährstoffe	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/nutritionalInformationModule/nutritionHeader/nutrientDetail/quantityContained	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	{ArticleSubDomainUOMType.LK.Std_LK_Text100_01#Nutrient basis quantity type selectable} , {ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Text100_01# Art der Bezugsgröße wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	ntity			
ArticleNutrientListQuantity.QuantityContainedUOM	Quantity contained UOM ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nutrient type selectable} , {ArticleSubDomainUOMType.LK.Std_LK_Text100_01#Nutrient basis quantity type selectable}	Enthaltene Menge: Maßeinheit ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nährstoffart wählbar}, {ArticleSubDomainUOMType.LK.Std_LK_Text100_01#Bezugsgröße wählbar}, {ArticleSubDomainUOMType.LK.UOMType	Nutrient quantity	Messwerte der Nährstoffe		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	, {ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	e#Einheitentyp wählbar}}				
ArticleNutrientListQuantity.DailyValueIntakePercent	Percentage of daily intake ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nutrient type selectable} , {ArticleSubDomainUOMType.LK.Std_LK_Text100_01#Nutrient basis quantity type selectable} , {ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Prozent des täglichen Bedarfs ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nährstoffart wählbar}, {ArticleSubDomainUOMType.LK.Std_LK_Text100_01#Art der Bezugsgröße wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar}}	Nutrient list quantity	Messwerte der Nährstoffe	nutritionalInformationModule/nutrientHeader/nutrientDetail/avpList/dailyValueIntakePercentMeasurementPrecisionCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleNutrientListQuantity.MeasurementPrecisionCode	Measurement precision ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nutrient type selectable} , {ArticleSubDomainUOMType.LK.Std_LK_Text100_01# Nutrient basis quantity type selectable} , {ArticleSubDomainUOMType.LK.UOMType# UOM type selectable})	Messgenauigkeit ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Nährstoffart wählbar}, {ArticleSubDomainUOMType.LK.Std_LK_Text100_01# Art der Bezugsgröße wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Nutrient list quantity	Messwerte der Nährstoffe	nutritionalInformationModule/nutritionHeader/nutrientDetail/measurementPrecisionCode	FAB
ArticleIngredient.IngredientOfConcernCode	Ingredient of concern code	Inhaltsstoff von Belang: Code	Ingredient in	Inhaltangaben	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeverageIngredientModule/ingredientOfConcernCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	M o d u l e (E n g l i s h)	Modu le (Ger man)	GDSN XML name	S c o p e
			f o r m a t i o n			
ArticleIngredient.JuiceContentPercentage	Juice content [%]	Saftgehalt [%]	In g r e d i e n t i n f o r m a t i o n	Inhalt sang aben	foodAndBeverageIngredientModule/juiceContentPercent	F A B
ArticleIngredient.Lang.IngredientStatement	Ingredient statement ({ArticleIngredientLangType.LK.Language#Language#LanguageSelectable})	Inhaltsstoffe ({ArticleIngredientLangType.LK.Language#Sprache wählbar})	L a n g u a g e - s p e z i f i c d	Sprachspezifische Daten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeverageIngredientModule/ingredientStatement	F A B

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ata			
ArticleIngredientAdditive.AdditiveName	Additive name	Zusatzstoff	Additives	Zusatzstoffe	foodAndBeverageIngredientModule/additiveInformation/additiveName	FAB
ArticleIngredientAdditive.LevelOfContainmentCode	Level of containment (ArticleSubDomainType.LK.Std_LK_Text250_01#Additive name selectable})	Grad des Vorkommens (ArticleSubDomainType.LK.Std_LK_Text250_01#Zusatzstoff wählbar})	Additives	Zusatzstoffe	foodAndBeverageIngredientModule/additiveInformation/LevelOfContainmentCode	FAB
ArticleIngredientComponent.Sequence	Sequence	Reihenfolge	Ingredient component	Inhaltsstoffe	foodAndBeverageIngredientModule/foodAndBeverageIngredient/ingredientSequence	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			nt			
ArticleIngredientComponent.Ingredient	Ingredient ({ArticleSubDomainType.LK.Std_LK_Text250_01#Sequence selectable})	Inhaltsstoff ({ArticleSubDomainType.LK.Std_LK_Text250_01#Reihenfolge wählbar})	IngredientComponent	Inhaltsstoffe		FAB
ArticleIngredientComponent.Purpose	Purpose ({ArticleSubDomainType.LK.Std_LK_Text250_01#Sequence selectable})	Zweck ({ArticleSubDomainType.LK.Std_LK_Text250_01#Reihenfolge wählbar})	IngredientComponent	Inhaltsstoffe	CatalogueItemNotification/ CatalogueItem/tradeItem/ tradeItemComponents/ foodAndBeverageIngredientModule/ foodAndBeverageIngredient/ ingredientPurpose	FAB
ArticleIngredientComponent.	Content [%] ({ArticleSubDomainType.L	Inhalt [%] ({ArticleSubDomainType.L	Ingredient	Inhaltsstoffe	foodAndBeverageIngredientModule/ foodAndBeverageIngredient/ ingredientContentPercentage	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ContentPercentage	bDomainType.LK.Std_LK_Text250_01#Sequence selectable}	K.Std_LK_Text250_01#Reihenfolge wählbar}}	diet component			
ArticleServingQuantity.MaximumNumberOfSmallestUnitsPerPackage	Maximum number of smallest units per package	Maximale Anzahl der kleinsten Einheiten pro Packung	Serving quantity information	Portionsmenge	foodAndBeveragePreparationServingModule/servingQuantityInformation/maximumNumberOfSmallestUnitsPerPackage	FAB
ArticleServingQuantity.NumberOfServingsPerPackage	Number of servings per package	Anzahl der Portionen pro Packung	Serving quantity	Portionsmenge	foodAndBeveragePreparationServingModule/servingQuantityInformation/numberOfServingsPerPackageMeasurementPrecisionCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			quantity information			
ArticleServingQuantity.MeasurementPrecisionOfNumberOfServingsPerPackage	Measurement precision of number of servings per package	Messgenauigkeit der Anzahl der Portionen pro Packung	Serving quantity information	Portionsmenge	foodAndBeveragePreparationServingModule/servingQuantityInformation/numberOfServingsPerPackageMeasurementPrecisionCode	FAB
ArticleServingQuantity.NumberOfSmallestUnitsPerPackage	Number of smallest units per package	Anzahl der kleinsten Einheiten pro Packung	Serving quantity	Portionsmenge	foodAndBeveragePreparationServingModule/servingQuantityInformation/numberOfSmallestUnitsPerPackage	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			quantity information			
ArticlePreparation.ManufacturerPreparationTypeCode	Manufacturer preparation type code	Zubereitungscode des Herstellers	Preparation Information	Zubereitungsangabe	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/foodAndBeveragePreparationServingModule/manufacturerPreparationTypeCode	FAB
ArticleNutritionalClaim.NutritionalClaimTypeCode	Nutritional claim type code	Nährwertanspruch: Code der Art	Nutritional claim	Nährwertanspruch	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/nutritionalInformationModule/nutritionalClaimDetail/nutritionalClaimCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ms			
ArticleNutritionalClaim.NutritionalClaimNutrientElementCode	Nutritional claim element code	Nährwertanspruch: Nährstoffcode	Nutritional claims	Nährwertanspruch	nutritionalInformationModule/ nutritionalClaimDetail/ nutritionalClaimNutrientElementCode	FAB
ArticleCertifications.Certification	Certification	Zertifizierung	Certification information	Zertifizierungsinformationen		FAB
ArticlePhysioChemical.PhysiochemicalCharacteristicCode	Physiochemical characteristic code	Physiochemische Eigenschaft	Physiochemical	Physiochemische Informationen	foodAndBeveragePropertiesInformationModule/ physiochemicalCharacteristic/ physiochemicalCharacteristicCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			mic al in f o r m a t i o n			
ArticlePhysioChemicalUOM.P hysiochemical Characteristic Value	Physiochemical characteristic value ({ArticleDomainUOM Type.LK.UOMType#UOM type selectable})	Wert der physiochemischen Eigenschaft ({ArticleDomainUOMType.L K.UOMType# Einheitentyp wählbar})	Physiochemical information UOM	Einheit der Physiochemischen Informationen	foodAndBeveragePropertiesInformationModule/ physiochemicalCharacteristic/ physiochemicalCharacteristicValue	F A B
ArticlePhysioChemicalUOM.P hysiochemical	Physiochemical characteristic value UOM	Einheit der physiochemischen Eigenschaft ({ArticleDomainUOMType.L K.UOMType# Einheitentyp wählbar})	Physiochemical information UOM	Einheit der Physiochemischen Informationen		F A B

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
Characteristic ValueUOM	{{ArticleDomainUOMType.LK.UOMType#UOM type selectable}}	inUOMType.LK.UOMType#Einheitentyp wählbar}}	chemical information UOM	chemische Informationen		
ArticleHealthCare.CompositionIncludeLatex	Does composition include latex	Enthält Latex	Health related information	Gesundheitsbezogene Informationen	healthRelatedInformationModule/healthRelatedInformation/doesTradeItemCompositionIncludeLatex	FAB
ArticleHealthCare.HealthClaimCode	Health claims	Code für gesundheitsbezogenen Angaben	Health related	Gesundheitsbezogene	healthRelatedInformationModule/healthRelatedInformation/healthClaimCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			lated information	Informationen		
ArticleHealthCare.IsNotIntendedForConsumption	Is not intended for consumption	Nicht zum Verzehr geeignet	Health related information	Gesundheitsbezogene Informationen	healthRelatedInformationModule/ healthRelatedInformation/ isTradeltemChemicalNotIntendedForHumanConsumption	F A B
ArticleHealthCare.NutritionLabelTypeCode	Nutrition label type code	Code der Nährwertkennzeichnung	Health related information	Gesundheitsbezogene Informationen	healthRelatedInformationModule/ healthRelatedInformation/ nutritionalLabelTypeCode	F A B

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			Information			
ArticleHealthCare.NutritionalProgramCode	Nutritional program code	Code des Ernährungsprogramms	Health related information	Gesundheitsbezogene Informationen	healthRelatedInformationModule/ healthRelatedInformation/ nutritionalProgramCode	FAB
ArticleHealthCare.Lang.CompulsoryAdditivesLabel	Compulsory additives label (ArticleDomainLangType.LK.Language#Language#selectable)	Verpflichtende Kennzeichnung von Zusatzstoffen (ArticleDomainLangType.LK.Language#Sprache wählbar))	Language specific	Sprachspezifische Daten	healthRelatedInformationModule/ healthRelatedInformation/ compulsoryAdditiveLabelInformation	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ata			
ArticleHealthCareLang.HealthClaimDescription	Description (ArticleDomainLangType.LK.Language#Language#Language#selectable)	Beschreibung (ArticleDomainLangType.LK.Language#Sprache#Sprache#wählbar)	Language specific data	Sprachspezifische Daten	healthRelatedInformationModule/ healthRelatedInformation/ healthClaimDescription	FAB
ArticleDairyFishMeatPoultry.FatInMilkContent	Fat in milk content [%]	Fett im Milchanteil [%]	Dairy fish meat poultry	Molkereiprodukte, Fisch, Fleisch, Geflügel	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ fatInMilkContent	FAB
ArticleDairyFishMeatPoultry.IsHomogenised	Is homogenised	Homogenisiert	Dairy fish	Molkereiprodukte,	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ isHomogenised/IsHomogenised	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
sHomogenised			shmeatpoultry	Fisch, Fleisch, Geflügel		
ArticleDairyFishMeatPoultryUOM.CasingTareWeight	Casing tare weight ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Verpackungsgewicht ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	DairyfishshmeatpoultryUOM	Maßeinheiten für Molkerieprodukte, Fisch, Fleisch, Geflügel	dairyFishMeatPoultryItemModule/dairyFishMeatPoultryInformation/casingTareWeight	FAB
ArticleDairyFishMeatPoultryUOM.CasingTareWeightUOM	Casing tare weight UOM ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Einheit des Verpackungsgewichts ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	DairyfishshmeatpoultryUOM	Maßeinheiten für Molkerieprodukte, Fisch, Fleisch, Geflügel		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ultray UOM	Geflügel		
ArticleCheese.CheeseMaturationProcessContainerTypeCode	Cheese maturation container process type	Gefäßtyp für den Käseerigungsprozess	Cheese information	Käseinformation	dairyFishMeatPoultryItemModule/dairyFishMeatPoultryInformation/cheeseMaturationProcessContainerTypeCode	FAB
ArticleCheese.FatPercentageInDryMatter	Fat in dry matter [%]	Fett in Trockenmasse [%]	Cheese information	Käseinformation	dairyFishMeatPoultryItemModule/dairyFishMeatPoultryInformation/fatPercentageInDryMatter	FAB
ArticleCheese.IsRindEdible	Is rind edible	Rinde verzehrbar	Cheese	Käseinformation	dairyFishMeatPoultryItemModule/dairyFishMeatPoultryInformation/isRindEdible	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ese information	mation		
ArticleCheese.RennetTypeCode	Rennet type code	Labart	Cheese information	Käseinformation	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ rennetTypeCode	FAB
ArticleCheese.SurfaceOfCheeseAtEndOfRipeningCode	Surface of cheese at end of ripening	Käseoberfläche am Ende der Reifezeit	Cheese information	Käseinformation	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ surfaceOfCheeseAtEndOfRipeningCode	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ion			
ArticleCheeseUOM.RipeningTimePeriod	Ripening time period (ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Reifezeit (ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Cheese information UOM	Einheit der Käseinformation	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ ripeningTimePeriod	FAB
ArticleCheeseUOM.RipeningTimePeriodUOM	Ripening time period UOM (ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Maßeinheit der Reifezeit (ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Cheese information UOM	Einheit der Käseinformation		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleCheeseLang.CheeseMaturationPeriodDescription	Cheese maturation period description ({ArticleSubDomainLangType.LK.Language#Language selectable})	Beschreibung des Zeitraums der Käsereifung ({ArticleSubDomainLangType.LK.Language#Sprache wählbar})	Language specific data	Sprachspezifische Daten	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ cheeseMaturationPeriodDescription	FAB
ArticleFishMeatPoultryContent.FishMeatPoultryTypeCode	Fish meat poultry type code	Fisch-, Fleisch-, Geflügelart	Fish meat poultry content	Fisch-, Fleisch-, Geflügelgehalt	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ fishMeatPoultryContent/ fishMeatPoultryTypeCode	FAB
ArticleFishMeatPoultryContent.FishMeatPoultryTypeCode	Fish meat poultry	Agentur zur Vergabe des Fisch-,	Fish	Fisch-, Fleisch	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
tryTypeCodeListAgency	type code list agency	Fleisch-, Geflügeltyps	meat poultry content	h-, Geflügelgehalt	fishMeatPoultryContent/ fishMeatPoultryTypeCodeListAgency	
ArticleFishMeatPoultryContent.FishMeatPoultryTypeCodeListIdentification	Fish meat poultry type code list identification	Identifikation des Fisch-, Fleisch-, Geflügeltyps	Fish meat poultry content	Fisch-, Fleisch-, Geflügelgehalt	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ fishMeatPoultryContent/ fishMeatPoultryTypeCodeListIdentification	FAB
ArticleFishMeatPoultryContentUOM.MinimumFishMeatPoultryContent	Minimum fish meat poultry content (ArticleSubDomainType	Minimaler Fisch-, Fleisch-, Geflügelanteil (ArticleSubDomainType.LK.	Fish meat poultry	Maßeinheiten des Fisch-, Fleisch	dairyFishMeatPoultryItemModule/ dairyFishMeatPoultryInformation/ fishMeatPoultryContent/ minimumFishMeatPoultryContent	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	pe.LK.Std_LK_Text25_01#Fish meat poultry type code selectable} , {ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Std_LK_Text25_01#Fisch-, Fleisch-, Geflügelart wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	oultry content unit UOM	h-, Geflügelgehalts		
ArticleFishMeatPoultryContentUOM.MinimumFishMeatPoultryContentUOM	Minimum fish meat poultry content UOM ({../ArticleSubDomainType.LK.Std_LK_Text25_01#Fish meat poultry type code selectable} , {ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Maßeinheit des minimalen Fisch-, Fleisch-, Geflügelanteils ({../ArticleSubDomainType.LK.Std_LK_Text25_01#Fisch-, Fleisch-, Geflügelart wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Fish meat poultry content unit UOM	Maßeinheiten des Fisch-, Fleisch-, Geflügelgehalts		FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.ReinstatementDate	Reinstatement date	Reinstatement date	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.NotLegalToAdvertise	Not legal to advertise by province	Not legal to advertise by province	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.NotLegalToDiscount	Not legal to discount by province	Not legal to discount by province	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.GSTHSTApplicable	GST/HST applicable	GST/HST applicable	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.PSTApplicableByProvince	PST applicable by province	PST applicable by province	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.EnvironmentalLevyApplicableByProvince	Environmental levy applicable by province	Environmental levy applicable by province	Canada-specific GDSN data	Canada-specific GDSN data	environmentalLevyApplicableByProvince	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.JuiceContentPercentage	Juice content [%]	Juice content [%]	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.MarkedLotNumber	Is marked lot number	Is marked lot number	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.SpecialHandlingCodeTransportation	Special handling code - transportation	Special handling code - transportation	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.ChannelOfDistribution	Channel of distribution	Channel of distribution	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.BrokerInvoiceIndicator	Broker invoice indicator	Broker invoice indicator	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.PercentageOfWaterContent	Percentage of water content	Percentage of water content	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.PercentageOfMoistureLoss	Percentage of moisture loss	Percentage of moisture loss	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.ProductionType	Production type	Production type	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.ExpiryDateFormat	Expiry date format	Expiry date format	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.ConsumerSupportNumber	Consumer support number	Consumer support number	Canada-specific GDSN data	Canada-specific GDSN data	consumerSupportNumber	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.IsCustomLabel	Is custom label	Is custom label	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.AverageServingsPerCase	Average servings per case	Average servings per case	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.ActiveIngredientGroupNumber	Active ingredient group number	Active ingredient group number	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.ActiveIngredientNameStrengthAndBasis	Active ingredient name strength and basis	Active ingredient name strength and basis	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.DosageFormType	Dosage form type	Dosage form type	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.DrugMerchandisingBtcProvince	Drug merchandising behind the counter by province	Drug merchandising behind the counter by province	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.Drug Merchandising OtcProvince	Drug merchandising OTC by province	Drug merchandising OTC by province	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.Drug Merchandising RxProvince	Drug merchandising Rx by province	Drug merchandising Rx by province	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.Drug Merchandising UnscheduledProvince	Drug merchandising unscheduled by province	Drug merchandising unscheduled by province	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.ShapeDescriptionFrench	Shape description french	Shape description french	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.TradeItemregulationTypeCodeControlledDrug	Trade item regulation type code controlled drug	Trade item regulation type code controlled drug	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.TradeItemregulationTypeCodeNarcotic	Trade item regulation type code narcotic	Trade item regulation type code narcotic	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtension.TradeItemregulationTypeCodePrecursor	Trade item regulation type code precursor	Trade item regulation type code precursor	Canada-specific GDSN data	Canada-specific GDSN data	-	IM
GDSNCanadaExtension.TradeItemregulationTypeCodeTargetedSubstance	Trade item regulation type code targeted substance	Trade item regulation type code targeted substance	Canada-specific GDSN data	Canada-specific GDSN data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNCanadaExtensionLang.OperatorDescription	Operator description ({ArticleDomainLangType.LK.Language#Language selectable})	Operator description ({ArticleDomainLangType.LK.Language#Language selectable})	Language-specific data	Language-specific data	-	IM
GDSNCanadaExtensionLang.ForMoreInfoAboutProduct	For more info about product ({ArticleDomainLangType.LK.Language#Language selectable})	For more info about product ({ArticleDomainLangType.LK.Language#Language selectable})	Language-specific data	Language-specific data	-	IM
GDSNCanadaExtensionLang.PackageAndStorage	Package and storage	Package and storage about product	Language-specific data	Language-specific data	packageAndStorageAboutProduct	IM

Product 360 Identifier	Display name (English)	Display name (German)	M o d u l e (E n g l i s h)	Module (German)	GDSN XML name	S c o p e
orageAboutPr oduct	about product ({ArticleDo mainLangT ype.LK.Lan guage#Lan guage selectable})	({ArticleDoma inLangType.L K.Language# Language selectable})	g u a g e- s p e c i f i c d a t a	fic data		
GDSNCanadaE xtensionLang. WeightScaleDe scription	Weight scale description ({ArticleDo mainLangT ype.LK.Lan guage#Lan guage selectable})	Weight scale description ({ArticleDoma inLangType.L K.Language# Language selectable})	L a n g u a g e- s p e c i f i c d a t a	Lang uage- speci fic data	-	I M
GDSNCanadaE xtensionLang. TradeItemMar kingsDescripti on	Trade item markings description ({ArticleDo mainLangT ype.LK.Lan guage#Lan	Trade item markings description ({ArticleDoma inLangType.L K.Language#	L a n g u a g	Lang uage- speci fic data	-	I M

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	language selectable})	Language selectable})	specific data			
GDSNCanadaExtensionUOM.SuggestedServingSize	Suggested serving size ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Suggested serving size ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	UOM	UOM	suggestedServingSize	IM
GDSNCanadaExtensionUOM.SuggestedServingSizeUOM	Suggested serving size UOM ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Suggested serving size UOM ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	UOM	UOM		IM
GDSNCanadaExtensionCustomer.Customer	Customer	Customer	Customer	Customer-specific data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			specific data			
GDSNCanadaExtensionCustomer.DeliveryLeadTime	Delivery lead time ({ArticleDomainPartyType.LK.Party#Customerselectable})	Delivery lead time ({ArticleDomainPartyType.LK.Party#Customerselectable})	Customer-specific data	Customer-specific data	-	IM
GDSNCanadaExtensionCustomer.MinimumTradeItemDaysInWarehouse	Minimum trade item days in warehouse ({ArticleDomainPartyType.LK.Party#Customerselectable})	Minimum trade item days in warehouse ({ArticleDomainPartyType.LK.Party#Customerselectable})	Customer-specific data	Customer-specific data	-	IM

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			data			
GDSNCustomerSpecificExtension.Party	Customer	Kunde	Customer - specific GDSN data	Kundenspezifische GDSN-Daten	-	GDSN
GDSNCustomerSpecificExtension.PublishItem	Publish item	Publiziere Artikel	Customer - specific GDSN	Kundenspezifische GDSN-Daten	-	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			Standard data			
GDSNTargetMarketExtension.IsActiveInMarket	Is active in market	Aktiv im Zielmarkt	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	-	GDSN
GDSNTargetMarketExtension.IsOrderableUnit	Is orderable unit	Bestelleinheit	Target market specific	Zielmarktspezifische GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradeItem/isTradeItemAnOrderableUnit	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			specific GDSN data			
GDSNTargetMarketExtension.IsDispatchUnit	Is dispatch unit	Liefereinheit	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradeItem/isTradeItemADispatchUnit	GDSN
GDSNTargetMarketExtension.IsInvoiceUnit	Is invoice unit	Fakturiereinheit	Target market	Zielmarktspezifische GDS	CatalogueItemNotification/CatalogueItem/tradeItem/isTradeItemAnInvoiceUnit	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			market specific GDSN data	N-Daten		
GDSNTargetMarketExtension.IsPackagingMarkedReturnable	Is packaging marked returnable	Verpackung Mehrwegkennzeichnung	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	packagingMarkingModule/ packagingMarking/ isPackagingMarkedReturnable	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ata			
GDSNTargetMarketExtension .Discontinued Date	Discontinued date	Auslaufdatum	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	CatalogueItemNotification/ CatalogueItem/tradeItem/ tradeItemSynchronisationDates/ discontinuedDateTime	GDSN
GDSNTargetMarketExtension .CountriesOfOrigin	Countries of origin	Ursprungsländer	Target market specific	Zielmarktspezifische GDSN-Daten	placeOfItemActivityModule/ placeOfProductActivity/ countryOfOrigin/countryCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			c G D S N d a t a			
GDSNTargetMarketExtension.PrimaryDeliveryMethod	Primary delivery method	Primäre Liefermethode	T a r g e t m a r k e t s p e c i f i c G D S N d a t a	Zielmarktspezifische GDSN-Daten	-	G D S N
GDSNTargetMarketExtension.UnitsPerCase	Total units per case	Einheiten pro Packung	T a r g e t m a r k	Zielmarktspezifische GDSN-Daten	-	G D S N

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			et s p e c i f i c G D S N d a t a	Daten		
ArticleMarketing.CouponFamilyCode	Coupon family code	Gutscheincod e	M a r k e t i n g i n f o r m a t i o n	Mark eting Infor mation	marketingInformationModule/ marketingInformation/ couponFamilyCode	G D S N
GDSNTargetMarketExtension.IsPrivate	Is private	Ist privat	T a r g e t m a r k e t s	Zielm arkts pezifi sche GDS N- Daten	isPrivate	G D S N

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			specific GDSN data			
GDSNTargetMarketExtension.PricingOnProduct	Pricing on product	Preisangabe am Produkt	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	packagingMarkingModule/ packagingMarking/isPriceOnPack	GDSN
GDSNTargetMarketExtension	Number of layers per pallet	Anzahl Lagen pro Palette	Target	Zielmarktspezifische	tradeItemHierarchyModule/ tradeItemHierarchy/ quantityOfLayersPerPallet	GD

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
.NumberOfLayersPerPallet			et market specific GDSN data	GDSN-Daten		SN
GDSNTargetMarketExtension .NumberOfItemsInPalletLayer	Number of items per pallet layer	Anzahl Artikel pro Palettenlage	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	tradeItemHierarchyModule/ tradeItemHierarchy/ quantityOfTradeItemsPerPalletLayer	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ata			
GDSNTargetMarketExtension.NumberOfItemsPerPallet	Number of items per pallet	Anzahl Artikel pro Palette	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	tradelItemHierarchyModule/tradelItemHierarchy/quantityOfTradelItemsPerPallet	GDSN
GDSNTargetMarketExtension.ProductMarkedRecyclable	Product marked recyclable	Ist recycelbar	Target market specific	Zielmarktspezifische GDSN-Daten	packagingMarkingModule/packagingMarking/isTradelItemMarkedAsRecyclable	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			c G D S N d a t a			
GDSNTargetMarketExtension.Manufacturer	Manufacturer	Hersteller	T a r g e t m a r k e t s p e c i f i c G D S N d a t a	Zielmarktspezifische GDSN-Daten	CatalogueItemNotification/ CatalogueItem/tradeItem/ manufacturer/gln; CatalogueItemNotification/ CatalogueItem/tradeItem/ manufacturer/partyName	G D S N
GDSNTargetMarketExtension.ReplacesGTIN	Replaces GTIN	Vorgänger-GTIN	T a r g e t m a r k	Zielmarktspezifische GDSN-	CatalogueItemNotificationType/ CatalogueItem/TradeItem/ dependentProprietaryTradeItem/ referencedTradeItemGTIN; CatalogueItemNotificationType/ CatalogueItem/TradeItem/ dependentProprietaryTradeItem/	G D S N

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			et s p e c i f i c G D S N d a t a	Daten	referencedTradelItemTypeCode = REPLACED	
GDSNTargetMarketExtension.ReplacedByGTIN	Replaced by GTIN	Nachfolger-GTIN	T a r g e t m a r k e t s p e c i f i c G D S N d a t a	Zielmarktspezifische GDSN-Daten	CatalogueItemNotificationType/CatalogueItem/TradelItem/dependentProprietaryTradelItem/referencedTradelItemGTIN; CatalogueItemNotificationType/CatalogueItem/TradelItem/dependentProprietaryTradelItem/referencedTradelItemTypeCode = REPLACED_BY	G D S N

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleMarketing.SpecialItem Code	Special item code	Sonderposten Code	Marketing information	Marketing Information	marketingInformationModule/ marketingInformation/ specialItemCode	GDSN
GDSNTargetMarketExtension.VariableTrade ItemTypeCode	Variable trade item type	Mengenvariabler Artikel: Art	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	variableTradeItemInformationModule/ variableTradeItemInformation/ variableTradeItemTypeCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNTargetMarketExtension.ChildNutritionLabel	Child nutrition label (USDA)	Kindernahrungsgtauglich (USDA)	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	catalogueItemNotification/catalogueItem/tradeItem/avpList/doesTradeItemCarryUSDACHildNutritionLabel	GDSN
GDSNTargetMarketExtension.IsDangerousSubstance	Is dangerous substance	Gefahrstoff	Target market specific GDSN	Zielmarktspezifische GDSN-Daten	dangerousSubstanceInformationModule/dangerousSubstanceInformation/dangerousSubstanceProperties/isDangerousSubstance	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			D S N d a t a			
GDSNTargetMarketExtension.IsRegulatedForTransportation	Is regulated for transportation	Regulierung für Transport	Target market specific GDSN data	Zielmarktspezifische GDSN-Daten	safetyDataSheetModule/safetyDataSheetInformation/isRegulatedForTransportation	GDSN
GDSNTargetMarketExtension.Lang.FunctionalName	Functional name ({ArticleMarketExtensionLangType.LK.Language#Language})	Funktionsbezeichnung ({ArticleMarketExtensionLangType.LK.Language#Sprache wählbar})	Language	Sprachspezifische Daten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/tradeItemDescriptionModule/tradeItemDescriptionInformation/functionalName	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	selectable})		specific data			
GDSNTargetMarketExtensionLang.ShortDescription	GDSN short description ({ArticleMarketExtensionLangType.LK.Language#Language selectable})	GDSN-Kurzbeschreibung ({ArticleMarketExtensionLangType.LK.Language#Sprache wählbar})	Language specific data	Sprachspezifische Daten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/tradeItemDescriptionModule/tradeItemDescriptionInformation/descriptionShort	GDSN
GDSNTargetMarketExtensionLang.AdditionalDescription	Additional description ({ArticleMarketExtensionLangType.LK.Language#Language selectable})	Zusätzliche Beschreibung ({ArticleMarketExtensionLangType.LK.Language#Sprache wählbar})	Language specific	Sprachspezifische Daten	tradeItemDescriptionModule/tradeItemDescriptionInformation/additionalTradeItemDescription	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			data			
ArticleMarketingLang.Market ingMessage	Marketing message ({ArticleDomainLangType.LK.Language#Language#Language#selectable} , {ArticleDomainLangType.LK.Std_LK_Int_01#Sequence#selectable})	Werbebotschaft ({ArticleDomainLangType.LK.Language#Sprache wählbar}, {ArticleDomainLangType.LK.Std_LK_Int_01#Reihenfolge wählbar})	Language specific data	Sprachspezifische Daten	marketingInformationModule/ marketingInformation/ tradeltemMarketingMessage	GDSN
GDSNTargetMarketExtension Lang.Variant	Variant ({ArticleMarketExtensionLangType.LK.Language#Language#Language#selectable})	Variation ({ArticleMarketExtensionLangType.LK.Language#Sprache wählbar})	Language specific data	Sprachspezifische Daten	tradeltemDescriptionModule/ tradeltemDescriptionInformation/ variantDescription	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNTargetMarketExtensionLang.ProductDescription	Product description ({ArticleMarketExtensionLangType.LK.Language#Language#Language selectable})	Artikelbeschreibung ({ArticleMarketExtensionLangType.LK.Language#Sprache wählbar})	Language specific data	Sprachspezifische Daten	tradeltemDescriptionModule/ tradeltemDescriptionInformation/ tradeltemDescription	GDSN
GDSNTargetMarketExtensionLang.SubBrand	Sub brand ({ArticleMarketExtensionLangType.LK.Language#Language selectable})	Submarke ({ArticleMarketExtensionLangType.LK.Language#Sprache wählbar})	Language specific data	Sprachspezifische Daten	tradeltemDescriptionModule/ tradeltemDescriptionInformation/ brandNameInformation/subBrand	GDSN
GDSNTargetMarketExtensionUOM.Height	Height ({ArticleMarketExtensionUOM.Height})	Höhe ({ArticleMarketExtensionUOM.Height})	UOM	Messwerte	tradeltemMeasurementsModule/ tradeltemMeasurements/height	GD

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	onUOMType.LK.UOMType#UOMtype selectable}	OMType.LK.UOMType#Einheitentyp wählbar}}				SN
GDSNTargetMarketExtensionUOM.HeightUOM	Height UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Maßeinheit der Höhe ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte		GDSN
GDSNTargetMarketExtensionUOM.Width	Width ({ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Breite ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte	tradeltemMeasurementsModule/tradeltemMeasurements/width	GDSN
GDSNTargetMarketExtensionUOM.WidthUOM	Width UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Maßeinheit der Breite ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNTargetMarketExtensionUOM.Depth	Depth (ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Tiefe (ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte	tradeltemMeasurementsModule/tradeltemMeasurements/depth	GDSN
GDSNTargetMarketExtensionUOM.DepthUOM	Depth UOM (ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Maßeinheit der Tiefe (ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte		GDSN
GDSNTargetMarketExtensionUOM.GrossWeight	Gross weight (ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Bruttogewicht (ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte	tradeltemMeasurementsModule/tradeltemMeasurements/tradeltemWeight/grossWeight	GDSN
GDSNTargetMarketExtensionUOM.GrossWeightUOM	Gross weight UOM (ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Maßeinheit des Bruttogewichts (ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	Type#UOM type selectable})	OMType#Einheitentyp wählbar})				
GDSNTargetMarketExtensionUOM.Netweight	Net weight ({ArticleMarketExtensionUOMType.LK.UOMType#UOM type selectable})	Nettogewicht ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte	tradeltemMeasurementsModule/tradeltemMeasurements/tradeltemWeight/netWeight	GDSN
GDSNTargetMarketExtensionUOM.NetweightUOM	Net weight UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOM type selectable})	Maßeinheit des Nettogewichts ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte		GDSN
GDSNTargetMarketExtensionUOM.DeliveryTemperatureMaximum	Delivery to distribution center temperature maximum ({ArticleMarketExtensionUOMType.LK.UOMType#UOM type	Transport zum Verteilzentrum: Temperatur (Max.) ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte	tradeltemTemperatureInformationModule/tradeltemTemperatureInformation/maximumTemperature;tradeltemTemperatureInformationModule/tradeltemTemperatureInformation/temperatureQualifierCode = DELIVERY_TO_DISTRIBUTION_CENTER	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	selectable})					
GDSNTargetMarketExtensionUOM.DeliveryTemperatureMaximumUOM	Delivery to distribution center temperature maximum UOM (ArticleMarketExtensionUOMType.LK.UOMType#UOMType selectable})	Maßeinheit für Transport zum Verteilzentrum: Temperatur (Max.) (ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar))	UOM	Messwerte		GDSN
GDSNTargetMarketExtensionUOM.DeliveryTemperatureMinimum	Delivery to distribution center temperature minimum (ArticleMarketExtensionUOMType.LK.UOMType#UOMType selectable})	Transport zum Verteilzentrum: Temperatur (Min.) (ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar))	UOM	Messwerte	tradeltemTemperatureInformationModule/ tradeltemTemperatureInformation/ maximumTemperature; tradeltemTemperatureInformationModule/ tradeltemTemperatureInformation/ temperatureQualifierCode = DELIVERY_TO_DISTRIBUTION_CENTRE	GDSN
GDSNTargetMarketExtensionUOM.DeliveryTemperatureMinimumUOM	Delivery to distribution center temperature minimum UOM (ArticleMa	Maßeinheit für Transport zum Verteilzentrum: Temperatur (Min.)	UOM	Messwerte		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	arketExtensionUOMType.LK.UOMType#UOMtype selectable}	{{ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}				
GDSNTargetMarketExtensionUOM.DeliveryToMarketTemperatureMaximum	Delivery to market temperature maximum {{ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable}}	Transport zum Markt: Temperatur (Max.) {{ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte	tradeltemTemperatureInformationModule/ tradeltemTemperatureInformation/ maximumTemperature; tradeltemTemperatureInformationModule/ tradeltemTemperatureInformation/ temperatureQualifierCode = DELIVERY_TO_MARKET	GDSN
GDSNTargetMarketExtensionUOM.DeliveryToMarketTemperatureMaximumUOM	Delivery to market temperature maximum UOM {{ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable}}	Maßeinheit für Transport zum Markt: Temperatur (Max.) {{ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar}}	UOM	Messwerte		GDSN
GDSNTargetMarketExtensionUOM.DeliveryToMarketTemperatureMaximum	Delivery to market temperature maximum	Transport zum Markt: Temperatur	UOM	Messwerte	tradeltemTemperatureInformationModule/ tradeltemTemperatureInformation/	GD

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
oMarketTemperatureMinimum	e minimum ({ArticleMarketExtensionUOMType.LK.UOMType#UOMType#UOMType selectable})	(Min.) ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})			maximumTemperature; tradeItemTemperatureInformationModule/tradeItemTemperatureInformation/temperatureQualifierCode = DELIVERY_TO_MARKET	SN
GDSNTargetMarketExtensionUOM.DeliveryToMarketTemperatureMinimumUOM	Delivery to market temperature minimum UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOMType selectable})	Maßeinheit für Transport zum Markt: Temperatur (Min.) ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte		GDSN
GDSNTargetMarketExtensionUOM.StorageHandlingTemperatureMaximum	Storage handling temperature maximum ({ArticleMarketExtensionUOMType.LK.UOMType#UOMType selectable})	Lagerungstemperatur (Max.) ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte	tradeItemTemperatureInformationModule/tradeItemTemperatureInformation/maximumTemperature; tradeItemTemperatureInformationModule/tradeItemTemperatureInformation/temperatureQualifierCode = STORAGE_HANDLING	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNTargetMarketExtensionUOM.StorageHandlingTempMaxUOM	Storage handling temperature maximum UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Maßeinheit der Lagerungstemperatur (Max.) ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte		GDSN
GDSNTargetMarketExtensionUOM.StorageHandlingTempMin	Storage handling temperature minimum ({ArticleMarketExtensionUOMType.LK.UOMType#UOMtype selectable})	Lagerungstemperatur (Min.) ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte	tradeItemTemperatureInformationModule/ tradeItemTemperatureInformation/ maximumTemperature; tradeItemTemperatureInformationModule/ tradeItemTemperatureInformation/ temperatureQualifierCode = STORAGE_HANDLING	GDSN
GDSNTargetMarketExtensionUOM.StorageHandlingTempMinUOM	Storage handling temperature minimum UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOMtype	Maßeinheit der Lagerungstemperatur (Min.) ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	selectable})					
GDSNTargetMarketExtensionUOM.Volume	Volume ({ArticleMarketExtensionUOMType.LK.UOMType#UOMType#selectable})	Volumen ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte	tradelItemMeasurementsModule/tradelItemMeasurements/inBoxCubeDimension	GDSN
GDSNTargetMarketExtensionUOM.VolumeUOM	Volume UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOMType#selectable})	Maßeinheit des Volumens ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte		GDSN
GDSNExtension.ProductType	Product type	Artikelebene	GDSN	GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradelItem/tradelItemUnitDescriptorCode	GDSN
GDSNExtension.BrandName	Brand name	Markenname	GDSN	GDSN-Daten	tradelItemDescriptionModule/tradelItemDescriptionInformation/brandNameInformation/brandName	GDSN
GDSNExtension.BrandOwner	Brand owner	Markenbesitzer	GDSN	GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradelItem/brandOwner/gln	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNExtension.IsService	Is service	Service-Einheit	GDSN	GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradeItem/isTradeItemAService	GDSN
GDSNExtension.IsConsumerUnit	Is consumer unit	Konsumenten einheit	GDSN	GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradeItem/isTradeItemAConsumerUnit	GDSN
GDSNExtension.IsBaseUnit	Is base unit	Basisartikel	GDSN	GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradeItem/isTradeItemABaseUnit	GDSN
GDSNExtension.HasVariableWeight	Has variable weight	Mengenvariabler Artikel	GDSN	GDSN-Daten	variableTradeItemInformationModule/variableTradeItemInformation/isTradeItemAVariableUnit	GDSN
GDSNExtension.QuantityOfNextLevelItems	Quantity of next level items	Gesamtzahl der Artikel der nächstniedrigeren Ebene	GDSN	GDSN-Daten	CatalogueItemNotification/CatalogueItem/tradeItem/nextLowerLevelTradeItemInformation / totalQuantityOfNextLowerLevelTradeItem	IM, data source
GDSNExtension.NumberOfLayersContainedInItem	Number of complete layers contained in item	Anzahl Lagen pro Palette	GDSN	GDSN-Daten	tradeItemHierarchyModule/tradeItemHierarchy/quantityOfCompleteLayersContainedInATradeItem	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNExtension.NumberOfItemsInALayer	Number of items in a complete layer	Anzahl Artikel in einer Lage	GDSN	GDSN-Daten	tradeltemHierarchyModule/ tradeltemHierarchy/ quantityOfTradelItemsContainedInACompleteLayer	GDSN
GDSNExtension.EffectiveDate	Effective date	Datum des Inkrafttretens	GDSN	GDSN-Daten	CataloguelItemNotification/ CataloguelItem/tradeltem/ tradeltemSynchronisationDates/ effectiveDateTime	GDSN
GDSNExtension.CanceledDate	Canceled date	Stornierungsdatum	GDSN	GDSN-Daten	CataloguelItemNotification/ CataloguelItem/tradeltem/ tradeltemSynchronisationDates/ cancelledDateTime	GDSN
GDSNExtension.GDSNRegistrationDate	GDSN registration date	GDSN-Registrierungsdatum	GDSN	GDSN-Daten	-	data Recipient
GDSNExtension.GDSNLastModifiedDate	Last modified date	Letzte Änderung am	GDSN	GDSN-Daten	-	data Recipient

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNExtension.ProductForm	Product form	Produktform	GDSN	GDSN-Daten	tradeltemDescriptionModule/ tradeltemDescriptionInformation/ tradeltemFormDescription	GDSN
GDSNExtensionLang.GTINName	GTIN name ({ArticleMarketExtensionLangType.LK.Language#Language#Language selectable})	GTIN-Name ({ArticleMarketExtensionLangType.LK.Language#Sprache wählbar})	Language specific GDSN data	Sprachspezifische GDSN-Daten	-	GDSN
GDSNExtensionUOM.NetContent	Net content ({ArticleMarketExtensionUOMType.LK.UOMType#UOMType selectable})	Nettofüllmenge ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte	tradeltemMeasurementsModule/ tradeltemMeasurements/netContent	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
GDSNExtensionUOM.NetContentUOM	Net content UOM ({ArticleMarketExtensionUOMType.LK.UOMType#UOMType selectable})	Maßeinheit der Nettofüllmenge ({ArticleMarketExtensionUOMType.LK.UOMType#Einheitentyp wählbar})	UOM	Messwerte		GDSN
ArticlePackaging.PackagingType	Packaging type	Verpackungsart	Packaging information	Verpackungsinformationen	packaginginformationModule/packaging/packagingTypeCode	GDSN
ArticlePackaging.PackagingFeatureCode	Packaging feature code	Verpackungseigenschaft	Packaging information	Verpackungsinformationen	packaginginformationModule/packaging/packagingFeatureCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ormation			
ArticlePackaging.PackagingFunctionCode	Packaging function code	Verpackungsfunktion	Packaging information	Verpackungsinformationen	packaginginformationModule/packaging/packagingFunctionCode	GDSN
ArticlePackaging.PlatformTypeCode	Platform type code	Palettenkennzeichen	Packaging information	Verpackungsinformationen	packaginginformationModule/packaging/platformTypeCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ion			
ArticlePackaging.PackagingShapeCode	Packaging shape code	Verpackungsform	Packaging information	Verpackungsinformationen	packaginginformationModule/ packaging/packagingShapeCode	GDSN
ArticlePackaging.DoesPackagingHaveWheels	Does packaging have wheels	Hat Verpackung Räder	Packaging information	Verpackungsinformationen	packaginginformationModule/ packaging/ doesPackagingHaveWheels	GDSN
ArticlePackaging.IsPackagingExemptFrom	Is packaging exempt	ARA entpflichtet	Packaging	Verpackungsin	packaginginformationModule/ packaging/	GD

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
RefuseObligation	from refuse obligation		packaging information	Verpackungsinformationen	isPackagingExemptFromRefuseObligation	SDN
ArticlePackagingRefuseObligationName	Packaging refuse obligation name	ARA Pflicht	Packaging information	Verpackungsinformationen	packaginginformationModule/ packaging/ packagingRefuseObligationNames	GDSN
ArticlePackagingPlatformTermsAndConditionsCode	Platform terms and conditions	Bedingungen der Palettennutzung	Packaging information	Verpackungsinformationen	packaginginformationModule/ packaging/ platformTermsAndConditionsCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			format ion			
ArticlePackagingUOM.UsableProductVolume	Usable product volume ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Nutzbare Produktvolumen ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Packaging information UOM	Messwerte der Verpackungsinformationen	packaginginformationModule/packaging/usableProductVolume	GDSN
ArticlePackagingUOM.UsableProductVolumeUOM	Usable product volume UOM ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Einheit des nutzbaren Produktvolumens ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Packaging information	Messwerte der Verpackungsinformationen		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			information UOM			
ArticlePackagingUOM.PackagingWeight	Packaging weight ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Verpackungsgewicht ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Packaging information UOM	Messwerte der Verpackungsinformationen	packaginginformationModule/ packaging/packagingWeight	GDSN
ArticlePackagingUOM.PackagingWeightUOM	Packaging weight UOM ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Einheit des Verpackungsgewichts ({ArticleDomainUOMType.LK.UOMType#UOM type selectable})	Packaging information UOM	Messwerte der Verpackungsinformationen		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			material UOM			
ArticlePackagingMaterial.PackagingMaterialTypeCode	Packaging material type code	Verpackungsmaterialtyp	Packaging material	Verpackungsmaterial	packaginginformationModule/packaging/packagingPackagingMaterial/packagingMaterialTypeCode	GDSN
ArticlePackagingMaterialUOM.PackagingMaterialCompositionQuantity	Packaging material composition quantity ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Packaging material type code selectable}, {ArticleSubDomainUOMType.LK.	Menge des Verpackungsmaterials ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Verpackungsmaterialstyp wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Packaging material UOM	Verpackungsmaterialmenge	packaginginformationModule/packaging/packagingPackagingMaterial/compositeMaterialDetail/packagingMaterialCompositionQuantity	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	UOMType# UOM type selectable})					
ArticlePackagingMaterialUOM.PackagingMaterialCompositionQuantityUOM	Packaging material composition quantity UOM (ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Maßeinheit des Verpackungsmaterials (ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Packaging material UOM	Verpackungsmaterialmenge		GDSN
ArticleGS1TradeItemIdentificationKey.GS1TradeItemIdentificationType	GS1 trade item identification type	Handelsgüteridentifizierungstyp (GS1)	GS1 trade item identification	Handelsgüteridentifizierungsschlüssel (GS1)	tradeItemDataCarrierAndIdentificationModule/ gs1TradeItemIdentificationKey/ gs1TradeItemIdentificationKeyCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			key			
ArticleGS1TradeItemIdentificationKey.GS1TradeItemIdentification	GS1 trade item identification	Handelsgüteridentifizierung (GS1)	GS1 trade item identification key	Handelsgüteridentifizierungsschlüssel (GS1)	tradeItemDataCarrierAndIdentificationModule/ gs1TradeItemIdentificationKey/ gs1TradeItemIdentificationKeyValue	GDSN
ArticleGS1TradeItemIdentificationKey.IsBarcodeSymbologyDerivable	Is barcode symbology derivable	Ist Barcode-Symbologie ableitbar	GS1 trade item identification	Handelsgüteridentifizierungsschlüssel (GS1)	tradeItemDataCarrierAndIdentificationModule/ gs1TradeItemIdentificationKey/ isBarcodeDerivable	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			notification key			
ArticleDeliveryPurchasing.StartAvailabilityDate	Start availability date	Verfügbarkeit: Startdatum	Delivery purchasing hash information	Lieferungs- und Einkaufsinformationen	deliveryPurchasingInformationModule/deliveryPurchasingInformation/startAvailabilityDateTime	GDSN
ArticleDeliveryPurchasing.EndAvailabilityDate	End availability date	Verfügbarkeit: Enddatum	Delivery purchasing	Lieferungs- und Einkaufsinformation	deliveryPurchasingInformationModule/deliveryPurchasingInformation/endAvailabilityDateTime	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			Archiving information	Archivierung		
ArticleDistributionDetail.IsDistributionMethodPrimary	Is distribution method primary ({ArticleSubDomainType.LK.Std_LK_Text250_01#Distribution method code selectable})	Primäre Vertriebsmethode ({ArticleSubDomainType.LK.Std_LK_Text250_01#Vertriebsmethode wählbar})	Distribution details	Vertriebsdetaile	deliveryPurchasingInformationModule/deliveryPurchasingInformation/distributionDetails/isDistributionMethodPrimary	GDSN
ArticleDistributionDetail.UOM. OrderingLeadTime	Ordering lead time ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Distribution method	Lieferzeit ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Vertriebsmethode wählbar})	Distribution details	Messwerte der Vertriebsdetaile	deliveryPurchasingInformationModule/deliveryPurchasingInformation/distributionDetails/orderingLeadTime	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	code selectable})		ais UOM			
ArticleDistributionDetailUOM. OrderingLeadTimeUOM	Ordering lead time UOM ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Distribution method code selectable})	Einheit der Lieferzeit ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Vertriebsmethode wählbar})	Distribution details UOM	Messwerte der Vertriebsdetails		GDSN
ArticleDataCarrier.DataCarrierFamilyType	Data carrier family type	Familienart des Datenträgers	Data carrier information	Datenträgerinformationen	tradeltemDataCarrierAndIdentificationModule/dataCarrier/dataCarrierFamilyTypeCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleDataCarrier.DataCarrierType	Data carrier type	Art des Datenträgers	Data carrier information	Datenträgerinformationen	tradeItemDataCarrierAndIdentificationModule/dataCarrier/dataCarrierTypeCode	GDSN
ArticleDataCarrier.ApplicationIdentifierType	Application identifier type	Application identifier	Data carrier information	Datenträgerinformationen	tradeItemDataCarrierAndIdentificationModule/dataCarrier/applicationIdentifierTypeCode	GDSN
ArticleDataCarrier.DataCarrierPresence	Data carrier presence	Datenträgerpräsenz	Data carrier	Datenträgerinformationen	tradeItemDataCarrierAndIdentificationModule/dataCarrier/dataCarrierPresenceCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			Carrier information			
ArticleTradeltemLifespan.DoesTradeltemHaveAutoReaderTracker	Does trade item have auto reader tracker	Automatische Verwendungsverfolgung verfügbar	Trade item lifespan	Handelsgegenstand	tradeltemLifespanModule/tradeltemLifespan/doesTradeltemHaveAutoReaderTracker	GDSN
ArticleTradeltemLifespan.MinProductLifespanFromArrival	Minimum product lifespan from arrival	Minimale Laufzeit des Artikels ab Wareneingang	Trade item lifespan	Handelsgegenstand	tradeltemLifespanModule/tradeltemLifespan/minimumTradeltemLifespanFromTimeOfArrival	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			an			
ArticleTradeltemLifespan.MinProductLifespanFromProduction	Minimum product lifespan from production	Minimale Laufzeit des Artikels ab Produktion	Trade item lifespan	Handelsgüterlebensdauer	tradeltemLifespanModule/tradeltemLifespan/minimumTradeltemLifespanFromTimeOfProduction	GDSN
ArticleTradeltemLifespan.OpenedTradeltemLifespan	Opened trade item lifespan	Laufzeit des Artikels ab Öffnung	Trade item lifespan	Handelsgüterlebensdauer	tradeltemLifespanModule/tradeltemLifespan/openedTradeltemLifespan	GDSN
ArticleTradeltemLifespan.SupplierSpecifiedMinimumConsumerStorage	Supplier specified minimum consumer storage	Minimale Laufzeit des Artikels ab Verkauf	Trade item	Handelsgüterlebensdauer	tradeltemLifespanModule/tradeltemLifespan/supplierSpecifiedMinimumConsumerStorageDays	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			mlifespan			
ArticleTradeltemHandlingUOM.ClampPressure	Clamp pressure (ArticleDomainUOMType.LK.UOMType#UOM type selectable)	Klammerdruck (ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar)	TradingUOM	Handelsgüterabfertigungseinheit	tradeltemHandlingModule/tradeitemHandlingInformation/clampPressure	GDSN
ArticleTradeltemHandlingUOM.ClampPressureUOM	Clamp pressure UOM (ArticleDomainUOMType.LK.UOMType#UOM type selectable)	Einheit des Klammerdrucks (ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar)	Trading	Handelsgüterabfertigungseinheit		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			UOM			
ArticleTradeltemHandlingLanguageHandlingInstructionsDescription	Handling instructions description ({ArticleDomainLangType.LK.Language#Language#Language#selectable})	Beschreibung der Abfertigungsanweisungen ({ArticleDomainLangType.LK.Language#Sprache wählbar})	Language specific data	Sprachspezifische Daten	tradeltemHandlingModule/tradeitemHandlingInformation/handlingInstructionsDescription	GDSN
ArticleTradeltemHandlingInstructions.HandlingInstructionCode	Handling instructions code	Abfertigungsanweisungen	Handling instructions	Abfertigungsanweisungen	tradeltemHandlingModule/tradeitemHandlingInformation/handlingInstructionsCodeReference	GDSN
ArticleTradeltemHandlingInstructions.Handl	Handling instructions code	Agentur für Abfertigungsanweisungen	Handl	Abfertigungsan	tradeltemHandlingModule/tradeitemHandlingInformation/	GD

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ingInstructionCodeAgency	agency ({ArticleSubDomainType.LK.Std_LK_Text250_01#Handling instruction code selectable})	({ArticleSubDomainType.LK.Std_LK_Text250_01#Abfertigungsanweisungen wählbar})	Handling instructions	Weisungen	handlingInstructionsCodeReferenceAgency	SN
ArticleTradeltemHandlingStacking.StackingFactorTypeCode	Stacking factor type code	Art des Stapelfaktors	Stacking instructions	Stapelung	tradeltemHandlingModule/tradeitemHandlingInformation/tradeitemStacking/stackingFactorTypeCode	GDSN
ArticleTradeltemHandlingStacking.StackingFactor	Stacking factor ({ArticleSubDomainType.LK.Std_LK_Text250_01#Stacking factor type code selectable})	Stapelfaktor ({ArticleSubDomainType.LK.Std_LK_Text250_01#Art des Stapelfaktors wählbar})	Stacking instructions	Stapelung	tradeltemHandlingModule/tradeitemHandlingInformation/tradeitemStacking/stackingFactor	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ns			
ArticleTradeltemHandlingStackingUOM.StackingWeightMaximum	Stacking weight maximum ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Stacking factor type code selectable} , {ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	Maximales Stapelgewicht ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Art des Stapelfaktors wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Stacking UOM	Einheiten der Stapelung	tradeItemHandlingModule/tradeItemHandlingInformation/tradeItemStacking/stackingWeightMaximum	GDSN
ArticleTradeltemHandlingStackingUOM.StackingWeightMaximumUOM	Stacking weight maximum UOM ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Stacking factor type code selectable} , {ArticleSubDomainUOMType.LK.	Einheit des maximalen Stapelgewichts ({../ArticleSubDomainType.LK.Std_LK_Text250_01#Art des Stapelfaktors wählbar}, {ArticleSubDomainUOMType.LK.UOMTyp	Stacking UOM	Einheiten der Stapelung		GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	UOMType# UOM type selectable}	e#Einheitentyp wählbar}}				
ArticleSustainability.DoesTradItemContainPesticide	Is product a pesticide	Ist das Produkt ein Pestizid	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/sustainabilityInformation/doesTradItemContainPesticide	DS E
ArticleSustainability.IsTradItemRigidPlasticPackagingContainer	Is product RPPC compliant	Ist das Produkt RPPC konform	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/sustainabilityInformation/isTradItemRigidPlasticPackagingContainer	DS E

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			ion			
ArticleSustainability.IsTradeltemROHSCompliant	Is product RoHS compliant	Ist das Produkt RoHS konform	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/sustainabilityInformation/isTradeltemROHSCompliant	DSE
ArticleSustainability.PostConsumerRecycledContentPercentage	Post consumer recycled material [%]	Recyclingmaterial nach Gebrauch [%]	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/sustainabilityInformation/postConsumerRecycledContentPercentage	DSE

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
ArticleSustainability.RenewablePlantBasedPlasticComponentsPercent	Plant based plastic components [%]	Pflanzlich hergestellte Kunststoffe [%]	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/sustainabilityInformation/renewablePlantBasedPlasticComponentsPercent	DS E
ArticleSustainability.ROHSComplianceFailureMaterial	Materials that fail RoHS compliance	Materialien die den Vorgaben der RoHS nicht entsprechen	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/sustainabilityInformation/rOHSComplianceFailureMaterial	DS E
ArticleSustainability.TotalRecycledContent	Total recycled content [%]	Gesamtmenge des	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/sustainabilityInformation/totalRecyclableContentPercentage	DS E

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
yclableContentPercentage		recycelten Inhalts [%]	availability information	Informa- tionen		
ArticleSustainability.TradeltemSustainabilityFeatureCode	Sustainability feature code	Nachhaltigkeitseigenschaft	Sustainability information	Nachhaltigkeitsinformationen	sustainabilityModule/ sustainabilityInformation/ tradeltemSustainabilityFeatureCode	D S E
Certification.isCertificateRequired	Is certificate required?	Ist ein Zertifikat erforderlich?	Certification	Zertifizierungs- informationen	certificationModuleInformation/ certificationInformation/ isCertificateRequired	G D S N

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			Information			
ArticleHealthCare.NutritionalScore	Nutritional Score	Nährwertangaben	Health related information	Gesundheitsbezogene Informationen	healthRelatedInformationModule/healthRelatedInformation/nutritionalScore	GDSN
ArticleDistributionDetail.DeliveryFrequencyCode	Delivery frequency code ({ArticleSubDomainType.LK.Std_LK_Text250_01#Distribution method code	Zustellhäufigkeitscode ({ArticleSubDomainType.LK.Std_LK_Text250_01#Vertriebsmethode wählbar})	Distribution details	Vertriebsdetai	deliveryPurchasingInformationModule/deliveryPurchasingInformation/distributionDetails/deliveryFrequencyCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	selectable})					
ArticleCheeseUOM.FatPercentageInDryMatterMeasurementPrecisionCode	Fat percentage in dry matter measurement precision code	Fettprozentatz in der Trockensubstanz Messgenauigkeitscode	Cheese information	Käseinformation	dairyFishMeatPoultryItemModule/dairyFishMeatPoultryInformation/fatPercentageInDryMatterMeasurementPrecisionCode	GDSN
ArticleHealthCare.Nutritional Value	Nutritional Value	Nährstoffgehalt	Health related information	Gesundheitsbezogene Informationen	healthRelatedInformationModule/healthRelatedInformation/nutritionalValue	GDSN
ArticleHealthCare.CannabisCBDTypeCode	Cannabis CBD Type Code	Cannabis CBD-Typ	Health related	Gesundheitsbezogene	healthRelatedInformationModule/healthRelatedInformation/cannabisCBDTypeCode	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			lated information	Informationen		
ArticleHealthCare.NutritionalProgramIngredientTypeCode	Nutritional Program Ingredient Type Code	Zutatentypen für das Ernährungsprogramm	Health related information	Gesundheitsbezogene Informationen	healthRelatedInformationModule/ healthRelatedInformation/ nutritionalProgramIngredientTypeCode	GDSN
ArticleHealthCareUOM.NutritionalProgramIngredientValue	Nutritional Program Ingredient Value	Inhaltsstoffmenge bzgl. des Ernährungsprogramms ({ArticleDomainUOMType.LK.UOMType#Einheitentyp wählbar})	Article Health Related Care	Gesundheitsbezogene Informationen Maßeinheit	healthRelatedInformationModule/ healthRelatedInformation/ nutritionalProgramIngredientValue	GDSN

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
			UOM			
ArticleHealthCareUOM.NutritionalProgramIngredientUnit	Nutritional Program Ingredient Unit	Einheit für Inhaltsstoffmenge bzgl. des Ernährungsprogramms ({ArticleDomainType.LK.UOMType# Einheitentyp wählbar})	Article Health Care UOM	Gesundheitsbezogene Informationen Maßeinheit	healthRelatedInformationModule/healthRelatedInformation/nutritionalProgramIngredientUnit	GDSN
ArticleDietList.IsDietTypeMarkedOnPackage	Is diet type marked on package ({ArticleSubDomainType.LK.Std_LK_Text250_01#Diet type code selectable})	Ist die Ernährungsart auf der Verpackung ausgewiesen ({ArticleSubDomainType.LK.Std_LK_Text250_01#Ernährungsart wählbar})	Diets	Ernährungsarten	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/dietInformationModule/dietInformation/dietTypeInfoInformation/isDietTypeMarkedOnPackage	FAB
ArticleNutrientListQuantity.ExpressedAsPartOf	Expressed as part of ({ArticleSubDomainType.LK.Std_LK_Text250_01#Nutrient type	Ausgewiesen als Teil von ({ArticleSubDomainType.LK.Std_LK_Text250_01#Nährstoffart wählbar},	Nutrient list	Messwerte der Nährstoffe	CatalogueItemNotification/CatalogueItem/tradeItem/tradeItemComponents/nutritionalInformationModule/nutrientHeader/nutrientDetail/expressedAsPartOf	FAB

Product 360 Identifier	Display name (English)	Display name (German)	Module (English)	Module (German)	GDSN XML name	Scope
	selectable} , {ArticleSubDomainUOMType.LK.Std_LK_Text100_01# Nutrient basis quantity type selectable} , {ArticleSubDomainUOMType.LK.UOMType#UOM type selectable})	{ArticleSubDomainUOMType.LK.Std_LK_Text100_01# Art der Bezugsgröße wählbar}, {ArticleSubDomainUOMType.LK.UOMType#Einheitentyp wählbar})	quantity			

2.1.8 GDSN Accelerator FAQ

2.1.8.1 General

Q: Do I need to take care on anything when I want to shut down Product 360?

A: Yes. If you need to shut down the Product 360 server for any reason (e.g. maintenance), it's advisable to also stop the PowerCenter workflows for GDSN. This is mainly due to the fact that you can still receive messages like CIC from the 1WorldSync datapool which will lead to a Service API call to Product 360 which will be created by the GDSN workflow. Hence all events with the "PIM update failed" status during the Product 360 downtime have to be reprocessed manually in the B2B DX web UI in case the PowerCenter workflows are still running. To avoid this manual step, stop the PowerCenter workflows before you shut down the Product 360 server and start them again after the Product 360 server is running again.

Q: Can I also send items to the GDSN pool from a supplier catalog?

A: This is theoretically possible but not recommended. The master catalog should contain the "golden" data which is send to the GDSN pool.

Q: What does the event status "PIM Update Failed" in the B2B Data Exchange mean?

<input type="checkbox"/>	Drill Up	Drill Down	Event ID	Partner	Account	Profile	Start Time	Event Type	Event Status	Aggregated Status
<input type="checkbox"/>			40149	Data Pool (GDSN)	pool_response_acc_name	prof_gdsn_message_from_data_pool (1102)	16 November 2016 15:55	CIC	PIM Update Failed	
<input type="checkbox"/>			40146	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	16 November 2016 10:54	CIP	Complete	Final

A: This means that the writing of the publication status in Product 360 wasn't successful. To evaluate what the problem was please click on the according Event ID and take a look in the "HTTP result report" file. You will find more information about the problem there.

Details of Event 40149

Event Details		Event Attributes	Event Status History	Reconciliation	Processing Information
Event ID	40149				Event status
Event Type	CIC				Start Time 16 November 2016 15:55:56.545
Partner	Data Pool (GDSN)				End Time 16 November 2016 15:56:09.496
Subject	Received: 2016-11-03-17-41-32-0217-as2				Duration 12 seconds, 951 milliseconds
Profile	prof_gdsn_message_from_data_pool				Aggregated Status

Event Logs

Log Type ▲	Date	Description	Doc Siz
Input	16 November 2016 15:55:56.576	Input message [2016-11-03-17-41-32-0217-as2.msg]	31
Intermediate	16 November 2016 15:55:56.557	This event is the new root of an event hierarchy.	1
Intermediate	16 November 2016 15:55:57.975	HTTP request	8
Intermediate	16 November 2016 15:56:08.180	HTTP result report	1

In following example there is no customer in Product 360 which has the given GLN:

```
{ "problems": [{"message": "Trade partner: The field value does not correspond to a value from the bullet Customers. ( 4322862000001[[]] )", "date": "2016-11-03-17-41-32-0217-as2"}]}
```

After adding the GLN to a customer please reprocess this message by selecting the event and choosing the action "Reprocess".

<input type="checkbox"/>	Drill Up	Drill Down	Event ID	Partner	Account	Profile	Start Time	Event Type	Event Status	Aggregated Status	Reconcile	Actions
<input checked="" type="checkbox"/>			40149	Data Pool (GDSN)	pool_response_acc_name	prof_gdsn_message_from_data_pool (1102)	16 November 2016 15:55	CIC	PIM Update Failed			Reprocess Resend Change event status... Release



The preference **automatedCustomerCreationEnabled** can be used to create unknown customers automatically during the REST-API call to Product 360.

Q: What does a response message like "GDSN Numeric Rule ID 1281: The format of "Ingredient Sequence" must be 'dd.dd.dd...'. Where 'd' must be a digit, always ending in a 'dd' and never having a value of '00'." mean?

A: This means the sent data is not valid regarding the validation which is done by the data pool. Usually the error message describes the deficiency in the data. You could add such a data check by a data quality rule configuration or a repository adjustment to ensure that you always send correct data.

2.1.8.2 Export

Q: The export fails due to the error "... One of '{document}' is expected."

ID	Type	Export post-process...	Validate XML file(s)	File 'Types.xsd' loaded
20	Note	Export post-process...	Validate XML file(s)	File 'Types.xsd' loaded
21	Note	Export post-process...	Validate XML file(s)	File 'AttrTypes.xsd' loaded
22	Mandato...	Export post-process...	catalogueRequest	File "ISync CatalogueRequest Item.xml": Error in line15, column 22: cvc-complex-type.2.4.b: The content of element 'catalogueRequest' is not complete. One of '{document}' is expected.
23	Summary	Post-export step	Validate XML file(s)	Export canceled due to XML validation error(s)

A: It can happen that there are no items in the generated XML file because no item passed the data quality checks or no item was passed to the export. If this case there will be no <document> tag generated in the XML which is leading to the above shown validation error. Despite to the fact that this is not very usable or obvious for an user, everything works correctly and there are no bad side effects. This behavior will be improved in a future Product 360 version.

Q: The export fails due to XSD error "...The value "" of attribute..."

Data type:
ID:
Position:

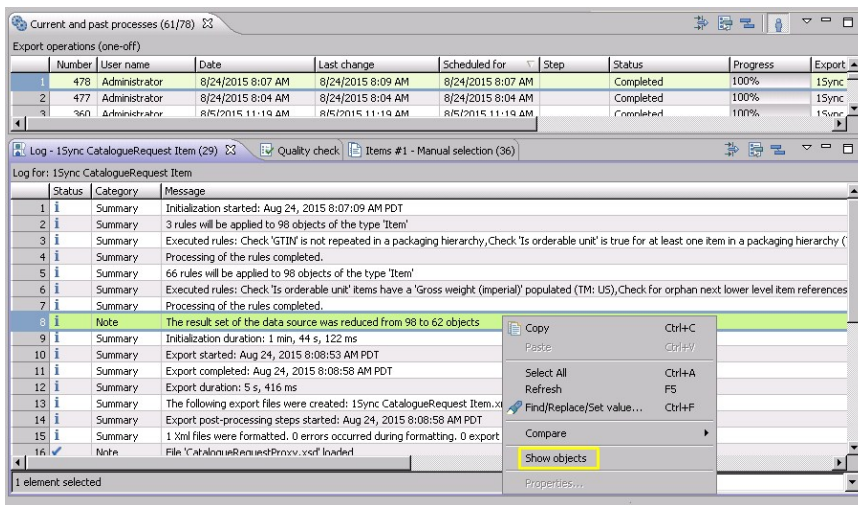
Message:

File "HF4EBF1_CatalogueItemRequest-MOD.xml": Error in line 1294, column 54: cvc-attribute.3: The value "" of attribute 'dataRecipientGLN' on element 'startAvailabilityDate' is not valid with respect to its type, 'glnType'.

A: If a generated XML file doesn't fit to the according GDSN XSD schema, an XSD validation error will occur. The above shown message indicates that no value was given for the attribute 'dataRecipientGLN' which is not valid regarding the 'glnType'. Therefore you need to open the "Customer" view and maintain the GLN for your customers.

Q: Not all of my items has been sent to the pool

A: Usually there is a data quality run before you send your data to the GDSN pool. If an item does not pass all data quality checks it will be filtered out and will not be sent to the GDSN pool. You can easily identify these items by navigating to the corresponding export log and select "Show objects" in the context menu of the corresponding log entry (see screenshot).



Q:Why does my GTIN differ in the exported file?

A:

When your GTIN is less than 14 digits in Product360 while exporting,the GTIN will be automatically filled with leading 0's to match the GS1 criteria of 14 digit GTINs. **It is strongly recommended to have 14 digit GTINs while working with Product360 GSDN Accelerator.**

Otherwise there will be mapping issues between the CICs received from from the pool.

2.1.8.3 B2B Data Exchange

Q: Can the existing B2B and PowerCenter implementation support more than one GDSN data pool?

A: No. Only one GDSN data pool can be supported with the current B2B and PowerCenter implementation. Additional implementation effort is needed in B2B Data Exchange and PowerCenter.

Q: Where is the Http connection for the REST API between PowerCenter and PIM configured?

A: The Http connection is configured in Workflow Manager tool. In the Menu bar, go to Connections > Application, choose to edit the Http Transformation called http_set_status, which was created during the B2B Data Exchange GDSN Accelerator installation.

Connection Object Definition

Application Connection Editor

Name: OK Cancel Help

Type:

User Name: ☐ Use Parameter In Password

Password:

Connect String:

Code Page:

Attributes:

Attribute	Value
Base URL	http://localhost:1512/rest/V1.0/manage/p...
Timeout	60
Domain	
Trust Certificates File	
Certificate File	
Certificate File Password	

Q: How can we check if the messages are processed and sent\received to\from the data pool correctly?

A: We can check the status of the B2B Data Exchange processes in various ways.

B2B Data Exchange Operator Console is a Web UI tool that is used to administer, configure and manage B2B Data Exchange. DX event is a representation of a file at a particular stage of processing. The B2B Data Exchange server generates events as it processes the files, and it changes the status of the events as they go through the transformation process.

The image below shows the Event List page. You can monitor and perform certain actions on events. You can perform a basic or advanced search for events and view event details.

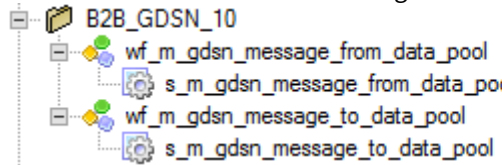




informatica B2B Data Exchange

Welcome: System Administrator | Logout

Event List									
Time frame: All (no range) Search Advanced search									
<input type="checkbox"/>	22069	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	28 July 2016 15:20	CIN	Complete	Final	Complete
<input type="checkbox"/>	22034	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	28 July 2016 10:18	CIN	Complete	Final	Complete
<input type="checkbox"/>	22032	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	28 July 2016 10:15	CIN	Complete	Final	Complete
<input type="checkbox"/>	22029	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	28 July 2016 09:12	CIN	Complete	Final	Complete
<input type="checkbox"/>	22010	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	28 July 2016 09:28	CIN	Complete	Final	Complete
<input type="checkbox"/>	19384	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	8 July 2016 13:52	CIC	Complete	Final	Complete
<input type="checkbox"/>	19380	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	8 July 2016 10:56	CIS	Complete	Final	Complete
<input type="checkbox"/>	19377	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	8 July 2016 10:52	CIS	Complete	Final	Complete
<input type="checkbox"/>	19373	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	8 July 2016 10:42	CIS	Complete	Final	Complete
<input type="checkbox"/>	19367	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	8 July 2016 10:40	CIN	Complete	Final	Complete
<input type="checkbox"/>	19363	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	8 July 2016 10:34	CIC	Complete	Final	Complete
<input type="checkbox"/>	19359	PIM	PIM_acc_name	prof_gdsn_message_to_data_pool (1101)	8 July 2016 10:31	CIC	Complete	Final	Complete

Another way to check if the files were processed successfully in PowerCenter is to check in Workflow Monitor tool:

1. Are GDSN related workflows running:

			
wf_m_gdsn_message_from_data_pool	00:00:34	Running	
s_m_gdsn_message_from_data_pool	00:00:34	Running	
wf_m_gdsn_message_to_data_pool	00:00:49	Running	
s_m_gdsn_message_to_data_pool	00:00:49	Running	

PowerCenter version 10.0, you can use Informatica MFT console to check if files were sent\received successfully to\from the data pool.

2. Check the session log of the workflow and search for "ERROR" messages using the "Severity" column:

Log Events for Session: s_m_gdsn_message_to_data_pool					
Severity	Timestamp	Node	Thread	Message Co...	Message
INFO	07.05.2017 15:23:39	node01_SUV1	DIRECTOR	TM_6014	Initializing session [s_m_gdsn_message_to_data_pool] at [Sun May 07 15:23:39 2017].
INFO	07.05.2017 15:23:39	node01_SUV1	DIRECTOR	TM_6683	Repository Name: [REPO_SVC_SUV12RDDEMO05]
INFO	07.05.2017 15:23:39	node01_SUV1	DIRECTOR	TM_6684	Server Name: [INT_SVC_SUV12RDDEMO05]
INFO	07.05.2017 15:23:39	node01_SUV1	DIRECTOR	TM_6686	Folder: [B2B_GDSN_10]
INFO	07.05.2017 15:23:39	node01_SUV1	DIRECTOR	TM_6685	Workflow: [wf_m_gdsn_message_to_data_pool] Run Instance Name: [] Run Id: [403]
INFO	07.05.2017 15:23:39	node01_SUV1	DIRECTOR	TM_6101	Mapping name: m_gdsn_message_to_data_pool.
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	TM_6964	Date format for the Session is [MM/DD/YYYY HH24:MI:SS.US]
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	CMN_65048	Recovery state file will be Synced to disk.
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	TM_6348	The maximum number of files that can be open at a time is 1024.
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	TM_6708	Using configuration property [EnableDataEncryption.no]
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	TM_6708	Using configuration property [StoreHAPersistenceInDB.no]
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	TM_6708	Using configuration property [IncludeErrorResponsesInHTTPTransformation.Yes]
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	TM_6708	Using configuration property [UseUTF8ForLocalHTTP.Yes]
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	TM_6703	Session [s_m_gdsn_message_to_data_pool] is run by 64-bit Integration Service [node01_SUV12RDDEMO05], version [10.1.0], build [0609].
INFO	07.05.2017 15:23:41	node01_SUV1	DIRECTOR	PETL_24088	Registering the session with the Integration Service

From PowerCenter version 10.0, you can use Informatica MFT console to check if files were sent\received successfully to\from the data pool.

- Verify if a requested file is sent successfully. Therefore check if there are errors in the logs of the configured AS2 server.
- Verify if there are errors in receiving a file. Therefore check the "Completed Jobs" logs.

2.1.9 GDSN implementation guidelines

2.1.9.1 Introduction

The scope of the GDSN implementation guidelines is to support adjustments of the provided GDSN Accelerator. This includes detailed technical instructions as well as things to consider. Furthermore it contains a list of things you need to clarify as a prerequisite, information about best practices when testing and information what to consider and think about if a different data pool should be supported.

To be able to use the GDSN implementation guidelines you need to have the following technical skills:

- You have a general understanding of GDSN.
- You are familiar with the Product 360 repository and you know about the available properties of the objects.
- You are able to create Product 360 import mappings and know how to use different import functionalities.
- You are able to export Product 360 data and know how to format the data correctly by using export functions.
- You are able to write Data Quality rules. (Optional, only needed if no standard DQ rule fits your needs.)
- You are able to configure Data Quality checks. (Optional)
- You have basic knowledge of B2B Data Exchange
- You have basic knowledge of Informatica PowerCenter

In addition the requirements for the needed GDSN attributes of your customer should be defined.

2.1.9.2 Documents

This documentation refers to already existing Product 360 documentation. Therefore you should have following documentation nearby:

- Previous chapters of GDSN Accelerator documentation
- Product 360 documents, which can be found at the Informatica partner webspace, especially the "Informatica MDM - Product 360 - <Version> - Knowledgebase, Installation and Customization.zip"
- Product 360 User Manual, which is part of the Product 360 delivery
- Data Quality documents, which are part of the Data Quality delivery

Furthermore this documentation refers to external documentation provided by your GDSN certified data pool. This includes documentation about

- the GDSN attributes (name, data type, description, valid value lists,...)
- valid value lists and their entries
- validations
- structure of the files used in the communication (if applicable)

The used documentations in this guideline are mentioned at the beginning of each chapter.

2.1.9.3 Analyze requirements

Introduction

This chapter describes how to analyze your GDSN requirements. First, you need to compare the needed GDSN attributes with the "[GDSN Accelerator field list\(see page 103\)](#)" to get a rough overview of how many new attributes need to be added. Please take care that you might need to adjust existing fields as well. This applies to new GDSN versions as well as implementing a new module and moving existing parts or fields respectively in the new module. As result of the analysis you should be able to call out which changes are needed and in case of data model changes, how the according entity and fields must be designed. Based on this analysis, the chapter "[Data model\(see page 235\)](#)" describes how to add new GDSN modules and GDSN attributes as well as creating or adjusting GDSN valid value lists.

In order to be able to analyze your customer's GDSN requirements, you need to have a good understanding of the Product 360 repository and the architecture it is based on.

Questions

The first step is to ask the right questions:

- Which scenario is used?
 - Data source or data recipient?
 - IM or DSE?
- Which modules are already available?
- What kind of change has to be implemented?
 - Is it a missing data field of an existing module?
 - Is it a missing module?
 - Is it a missing validation?

- Is it a missing value of a valid value list?
- ...



You can find all supported GDSN attributes by the GDSN Accelerator in the chapter "[GDSN Accelerator field list](#)(see page 103)".

Resources

It is essential to have all necessary documents available. These are the GDSN specification documents you can get from the corresponding GS1 home page.

There are several PDF and Excel files describing fields, data types, validations. And there are XSD files you should use to get detailed information about the structure of the GDSN message files, especially of the files containing item data.

In the following examples we use these files:

- XML Schemas
- IM_Participant_Dictionary_R7.1.0_v1 (especially tabs IM Participant dictionary and IM Valid Values)
- GDSN module PDF files
- Data_Source_1WS_XML_Guide_IM7.0v6.pdf called "Data Source XML Guide" in the rest of the document
- IM_Validations_Document_R7.1.0_v1.xlsx

Data model - analyze the module

So let's work together on a first "module"... We've been told by the customer, that he needs to store microbiological information and wants to make it available using *GDSN IM*. Where do we start?

Collect information

The customer is a manufacturer and uses the *data source scenario*, so let's have a look at "1WorldSync Item Management - Data Source 1WS XML Guide".

When we search for "Microbiological" we first find this line

1WS Catalogue Request Attribute	1WS XML Structure Type	1WS Item Structure Type	Module	FLX
foodAndBevMicrobiological	AGM	O	FoodAndBeveragePropertiesInformation	Y

What does it mean?

1WS Catalogue Request Attribute: This means, that a structure element with the name `foodAndBevMicrobiological` exists. The structure type is "AGM" which stands for "attribute group many" and indicates that it contains a group of attributes and can occur multiple times in the XML of the Catalogue Request file that is sent to the 1WS Pool.

1WS XML Structure Type: The structure type "AGM" corresponds to the information "FLX: Y" which means that the attribute is a flex attribute. Flex attributes are a generic way to include attributes in the XML structure.

This is an example of how a flex attribute will look like in the XML of the Catalogue Request file:

Flex attribute

```
<flex>
  . . .
  <attrGroupMany name="shipFromPartyInformation">
    <row>
      <attr name="glnOfShipFromParty">6701115112308</attr >
      <attr name="nameOfShipFromParty" qual="USD">GLN Name</attrQual>
    <row>
      <row>
        <attr name="glnOfShipFromParty">6701115112308</attr >
        <attr name="nameOfShipFromParty" qual="USD">GLN Name</attrQual>
      <row>
    </attrGroupMany>
  . . .
</flex>
```

Here's a list of other structure types, just to give you an idea:

- A - attribute
- AM - attribute many
- AQ - attribute qualified
- AGM - attribute group many

For further information on flex attributes and structure types see explanations and examples in "1WorldSync Item Management - Data Source 1WS XML Guide".

1WS Item Structure Type: The "O" tells us, that the attribute group is optional.

Module: In the "Module" column it says "FoodAndBeveragePropertiesInformation", so there is no separate module for microbiological information. We should have a look at what else is contained in the "FoodAndBeveragePropertiesInformation" module in order to decide how to design our entities. We note that down and go on for now.

Next we find four lines that seem to be fields

1WS Catalogue Request Attribute	1WS XML Structure Type	1WS Item Structure Type	1WS Item Data Type	1WS Item Data Length (Min)	1WS Item Data Length (Max)	Qualifier	Module	FLX
foodAndBevMicrobiological/organismCode	A	O	string	1	80		FoodAndBeveragePropertiesInformation	Y
foodAndBevMicrobiological/organismMaximumValue	AQ	O	ufloat	15	15	uom	FoodAndBeveragePropertiesInformation	Y
foodAndBevMicrobiological/organismReferenceValue	AQ	O	ufloat	33	2	uom	FoodAndBeveragePropertiesInformation	Y
foodAndBevMicrobiological/organismWarningValue	AQ	O	ufloat	33	2	uom	FoodAndBeveragePropertiesInformation	Y

What information do we get here?

In the group "foodAndBevMicrobiological" there are four fields

1. The attribute "organismCode" which is a simple string attribute with a max. length of 80 that is optional.
2. The qualified attribute "organismMaximumValue" which is a decimal value with a min. length of 15 and a max. length of 15. The Qualifier "uom" is an indicator that we need another field to store the value for the qualifier. In some cases, like for example the language, the qualifier can be a logical key.
3. and 4. are the qualified fields "organismReferenceValue" and "organismWarningValue", both qualified with a unit, both decimal, both optional, both with a min. length of 33 and a max. length of 2. Well, the length can't be correct. We should check that later in the Participant dictionary. Write that down and go on.



"Uom" stands for "unit of measure". We use the terms "uom" and "unit" interchangeably.

Measurement values always consist of a pair of value and uom.

If we keep searching we find these lines

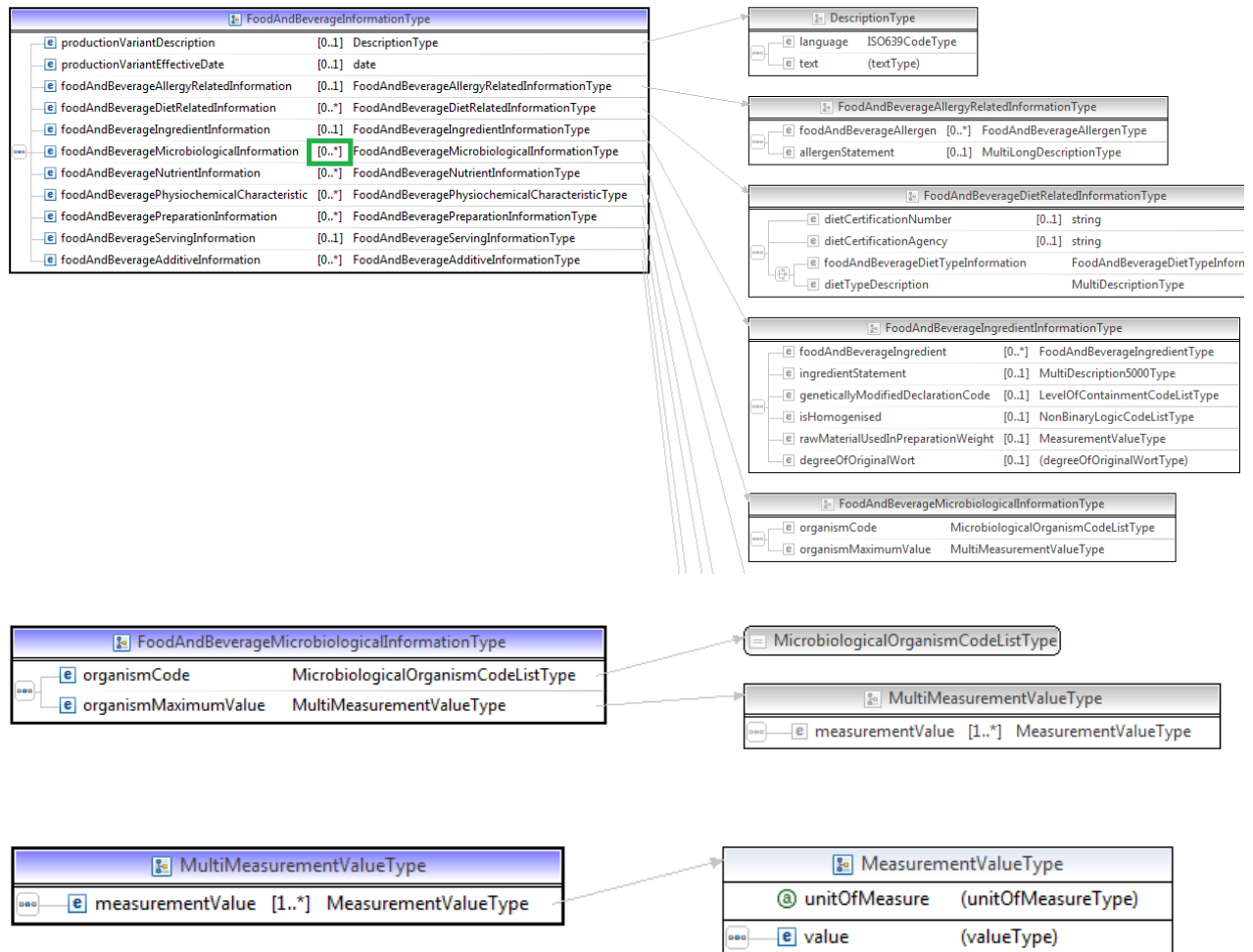
1WS Catalogue Request Attribute	1WS XML Structure Type	1WS Item Structure Type	1WS Item Data Type	1WS Item Data Length (Min)	1WS Item Data Length (Max)	Qualifier	Module	FLX
componentInformation/ foodAndBeveragePropertiesInformation/ foodAndBevMicrobiological/ organismCode	A	O	string	1	80		ComponentInformation	Y
componentInformation/ foodAndBeveragePropertiesInformation/ foodAndBevMicrobiological/ organismMaximumValue	AQ	O	ufloat	15	15	uom	ComponentInformation	Y
componentInformation/ foodAndBeveragePropertiesInformation/ foodAndBevMicrobiological/ organismReferenceValue	AQ	O	ufloat	15	15	uom	ComponentInformation	Y
componentInformation/ foodAndBeveragePropertiesInformation/ foodAndBevMicrobiological/ organismWarningValue	AQ	O	ufloat	15	15	uom	ComponentInformation	Y

If you compare this set of fields with the ones we found before you will notice that the attribute names are the same but the paths are different. The module is different, too. It's "ComponentInformation".

Components are a Mjr3 feature that we don't support at the moment, so ignore these fields. If you want to know more about components have a look at the GDSN homepage (<http://www.gs1.org/gdsn>).

Get a better idea of the structure

Since the fields we found are flex attributes, we won't see much of them in the IM XSDs. Sometimes it's hard to imagine the complete structure of a module based only on the textual information given in the table. But we can get a little help from the GDSN XSDs to get a better idea how the structure might look like.



If we look at the "FoodAndBeverageInformationType", marked in the green we see the occurrences are "0..*", so we can have multiple entries for "FoodAndBeverageMicrobiologicalInformation". This means we have to find a logical key for our data model. We have the organismCode. Since the rest are measurement values, it seems to make sense to have one set of values for each organism. So this is a good candidate for a logical key.

In GDSN there is only the "organismMaximumValue". We also found that field in the "1WorldSync Item Management - Data Source 1WS XML Guide" but additionally there were "organismReferenceValue" and "organismWarningValue". In the XSD we can see that there is an additional "unitOfMeasure" belonging to the

"organismMaximumValue". This is no surprise, we already saw that the fields are qualified with a unit and knew that we needed to store this information somewhere.

So let's recap:

We have the "foodAndBeverageMicrobiological" module, which is an attribute group many. This sounds like an entity, doesn't it?

We have identified the fields

- organismCode
- organismMaximumValue
- organismMaximumValueUOM
- organismReferenceValue
- organismReferenceValueUOM
- organismWarningValue
- organismWarningValueUOM

We also know that "organismCode" is a candidate for a logical key.

Design the entity

You don't know yet, but it will be described in the section "[Data model\(see page 235\)](#)" that there is the entity type `ArticleDomainType` that is suitable for implementing new modules and that has a sub entity for measurement values. There is an additional key `UOMType`. Possible values are: `METRIC` and `IMPERIAL`. We need to be able to store multiple values, but since units are convertible we don't need to store each and every value we might be using in an output. That's the reason we don't use the unit as the logical key.

So what we have to do is to create an entity like this:

- Entity: `ArticleMicrobiological` (based on `ArticleDomainType`)
 - Logical key: `organismCode`
 - Field: `organismCode`
 - Entity: `ArticleMicrobiologicalUOM` (based on `ArticleDomainUOMType`)
 - Logical key: `UOMType`
 - Field: `UOMType`
 - Field: `organismMaximumValue`
 - Field: `organismMaximumValueUOM`
 - Field: `organismReferenceValue`
 - Field: `organismReferenceValueUOM`
 - Field: `organismWarningValue`
 - Field: `organismWarningValueUOM`

Now, we have the basic structure of our sub entity.

Check the details

Now, we have to check for the details. The details can be found in the Participant dictionary.

	C	D	E	F	G	H	I	J	K	L	M	N	O
	GSI Location	GSI Name	IM XML Name	GDSN XML Name	Mandatory/Optional	Definition	Data Type	Length (All Rights), Min Length (All Non-Rights Data Types)	Precision (All Rights), Max Length (All Non-Rights)	Global/Target Market Specific	Occurrence	Qualifier Type	Qual Valid Value List
1	Components Tab	Organism Code	componentInformation/foodAndBeveragePropertiesInformation/foodAndBevMicrobiological/organismCode	CatalogItemNotification/CatalogItem/tradeItem/tradeItemComponents/foodAndBeveragePropertiesInformation/foodAndBevMicrobiologicalInformation/microbiologicalOrKeratinCode	O	Code indicating the type of microbiological organism.	VV/FNBOrganismCode	1	80	Target Market	0..1		
547	Components Tab	Organism Maximum Value	componentInformation/foodAndBeveragePropertiesInformation/foodAndBevMicrobiological/organismMaximumValue	CatalogItemNotification/CatalogItem/tradeItem/tradeItemComponents/foodAndBeveragePropertiesInformation/foodAndBevMicrobiologicalInformation/microbiologicalOrKeratinMaximumValue	O	Maximum allowable value of the microbiological organism.	ufloat	15	15	Target Market	0..1	uom	uom
548					O								

Search for the attributes we found earlier:

foodAndBevMicrobiological/organismCode

GUI Name: Organism Code

This is your English display label. By convention in Product 360, the first letter of the first word is upper case, all following words start with a lower case character except it is a name or another word which is correctly spelled with an upper case character in English.

IM XML Name: foodAndBevMicrobiological/organismCode

This is where you find the path given in the Data Source XML Guide

GDSN XML Name:

Where you find it in the XSDs of the GDSN XML structure.

Mandatory/Optional:

Most of the fields are optional, some of the fields are "M within O group" which means they are mandatory if you form the group. These fields can be set to mandatory in the repository (upper and lower bound = 1). Check if these fields are suitable as logical keys.

Definition:

This is your description in English. Read it, correct it, if it is a whole sentence add a '.' at the end and if there is no valuable descriptive content in there, don't use it.

Datatype: VV/FNBOrganismCode

"VV" means there is a valid value list. "FNBOrganismCode" is the name of the list. The values will be found on the tab "IM Valid Values". Valid value lists most likely contain strings.
Other common data types apart from valid value lists are dateTime, uinteger and ufloat.
→ We need to add an enumeration to the repository as well.

Min. length and max. length:

The field can have values of strings with a length up to 80. Even they say the min. length is 1, since the value is optional I would use a lower bound of 0.

Global/Target market specific: Target Market

Defines if it is possible to maintain different values for different target markets.
→ We need an additional target market key.

Occurrence: 0..1

This can be a problem if we want to use organism code as a logical key. Logical keys are mandatory.

foodAndBevMicrobiological/organismMaximumValue

Most of the information is similar to the above.

The data type is ufloat and the columns of the min./max. length are called "**Length (All floats)**, Min. Length (All non-float Data Types)" and "**Precision (All floats)**, Max. Length (All non-float)". This explains the values 33/2 which made no sense earlier.

Example: Imagine an attribute defined with 15/15. What this means is that 1234567890,12345 is valid and 12345,1234567890 is valid but 12345678,12345678 is not valid because the complete length is greater than 15.

Product 360 can't persist such huge numbers. A BigDecimal16/6 is always used which means the complete number is at most 16 places long - 10 places before the decimal separator and 6 decimal places. However, if the definition is for example 5/2 the max. range should be set to 99999,99 with scale 2. Be aware that this means you can store 88888,888888 in the database anyway because the scale is just a matter of formatting.

foodAndBevMicrobiological/organismReferenceValue and foodAndBevMicrobiological/organismWarningValue

Most of the information is similar to the above.

Have a look at the length and precision. Here in the Participant dictionary it says 15/15 not 33/2 as it did in the Data Source XML Guide. This is an example of conflicting documentation. Note this on your test list and send dummy data to the data pool later. Determine what is correct on the error messages you get back from the data pool.

UOM

We know we need the unit fields as well. We won't find them as separate lines in the participant dictionary, only as qualifier in the line of the attribute they belong to. At a first glance you might wonder how you will be able to create the field with so little information. However since units will always be stored as `UnitProxies` and corresponding field types are already provided in the `ArticleDomainUOMType`, there is not much left to be configured.

The only question we have to answer is, which units should be available to the user or in other words which enumeration do we need to add to the unit field. If we go back to the Participant dictionary there is no VV entry in the datatype column, which makes sense because this line is about the measurement value that is a numeric value.

GUI Name	IM XML Name	DataType	Length (All floats), Min Length (All non-float Data Types)	Precision (All floats), Max Length (All non-float)	Qualifier Type	Qual Valid Value List
Organism Warning Value	componentInformation/foodAndBeveragePropertiesInformation/foodAndBevMicrobiological/organismWarningValue	ufloat	15	15	uom	uom

But there are two other columns which will give us the answer we need. There is the qualifier type "uom" and "Qual Valid Value List" "uom". If you go to the tab "IM Valid Values" you will find a list with that name. In the "Attribute Name" column of that tab you will also find the "organismWarningValue" attribute and the other value fields. Check the existing enumerations if there is already one with the matching values or create your own. See section "[Data model\(see page 235\)](#)" for information how to do that.



It has proven to be useful to create an excel sheet with all the information relevant to you. This may include:

- GDSN attribute name
- PIM display label
- Field identifier

- Data type in GDSN
- Data type in PIM
- Valid values
- Is field mandatory?
- ...

Logical keys

Let's come back to the hardest decision. What do we use as logical key(s)?

- *Do we need a logical key?*
Yes, foodAndBevMicrobiological is AGM, so we need to be able to store more than one set of values per target market.
- *Why should we use organismCode as logical key?*
It makes sense to have one set of measurement values per organism. It doesn't make much sense to have multiple warning values for the same organism from a business point of view.
- *Why shouldn't we use organismCode as logical key?*
Because organismCode is optional. This means GDSN allows to have measurement values not belonging to one of the organisms in the valid values list.
- *Is there an alternative?*
Can't think of one.

What is the solution then?

The solution is to use "organismCode" as logical key but tweak the valid values a little. Make an enumeration "with optional code". This enumeration has one or more additional values. Most standard enumerations with optional code have one additional value, for example "NONE". This allows the user to store additional value sets, it works with all the generic mechanisms in Product 360 and in the export there is a mechanism which will ensure that this value is not sent to the GDSN data Pool. How many additional entries (if at all) you need depends on the requirements of the customer.

See how to create an enumeration with optional code in the chapter "[Data model\(see page 235\)](#)".

See how to handle enumerations with optional codes in the export in section "[Technical details\(see page 84\)](#)".

Compacting the structure

At the beginning we saw that the attributes related to microbiological information belong to the module "FoodAndBeveragePropertiesInformation". We ignored that up to now. But you might ask yourself if you have to fit a complete module into one Article sub entity.

The answer is definitely 'no'.

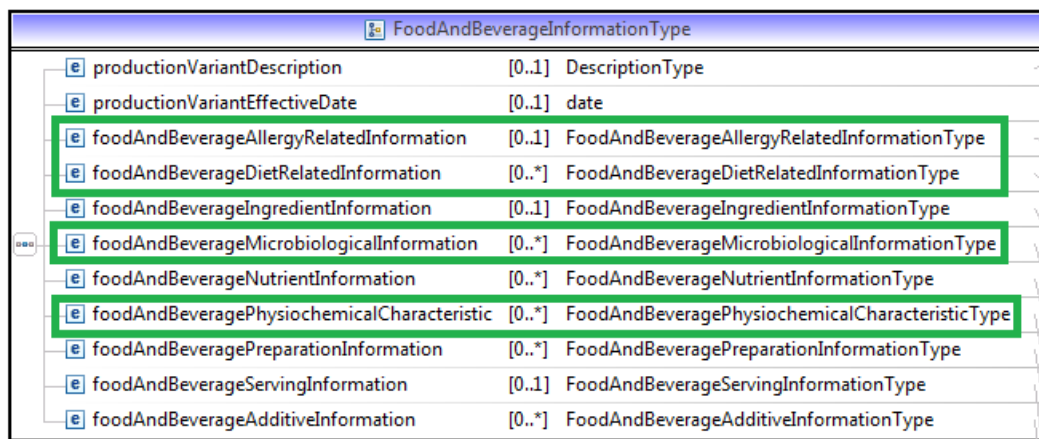
Let's see what else is contained in the module "FoodAndBeveragePropertiesInformation": What we find are physiochemical properties.

1WS Catalogue Request Attribute	1WS XML Structure Type	1WS Item Structure Type	Module	FLX
physioChemicalProperties	AGM	0	FoodAndBeveragePropertiesInformation	Y

1WS Catalogue Request Attribute	1WS XML Structure Type	1WS Item Structure Type	1WS Item Data Type	1WS Item Data Length (Min)	1WS Item Data Length (Max)	Qualifier	Module	FLX
physioChemicalProperties/ physioChemicalCharacteristic Code	A	O	string	1	80		FoodAndBeveragePropertiesInformation	Y
physioChemicalProperties/ physioChemicalCharacteristic Value	AQM	O	ufloat	15	15	uom	FoodAndBeveragePropertiesInformation	Y

Does this information have a relation to microbiological information?

No. So we probably can implement the physiochemical properties in its own sub entity. When we look at the GDSN XSDs, we get a confirmation of our assumption. FoodAndBeverage**Microbiological**InformationType and FoodAndBeverage**PhysioChemical**CharacteristicType are two separate types on the same level as FoodAndBeverage**AllergyRelated**Information and FoodAndBeverage**DietRelated**Information.



Deep structures

Some modules, for example the ingredient information, have a pretty deep structure with up to ~ 10 nested levels of XML tags. The ArticleDomainType has a depth of 3 (+1 for the item itself).

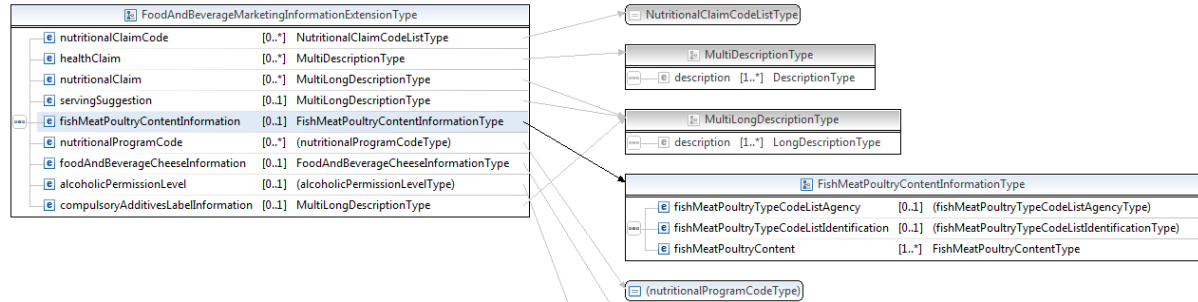
If you encounter a module with a deep structure you have to get creative and shrink it down to 3 levels.

To get you started here are two possible ways to do that:

1. Extraction

Certifications are an example of Extraction. You can add a certification for the item itself and you can add a certification for a specific diet. The XML sub structure that stores the certification information is the same in both cases. Furthermore if you think object oriented, a certification is a self-contained complete object. So in Product 360 that part was extracted into its own root entity and only the entity proxy is stored in the corresponding sub entities of the item.

2. Compacting



As you can see "fishMeatPoultryContentInformation" is contained in "FoodAndBeverageMarketingInformationExtension". But it has no relation to the rest of the information in the marketing information. This is a case where the container is not really needed for the logical structure of the information and we can skip this layer. The occurrence of "0..1" confirms that no sub entity layer with logical keys is needed at this point.

Ways to ensure data consistency

In the course of creating the data model, you should start thinking about data consistency or in other words validations.

There are two sources for information about validations. The Participant dictionary contains a lot of basic validations like the max. field length. Complex validations, you have to take into account for the design of your entity, are listed in the "IM Validations Document".

Data model

There are lower level validations only affecting a single field that can be configured in the repository.

Valid value lists

Description: Some fields only allow a certain set of values.

Where to find: Information can be found in the Participant dictionary in column "DataType", for qualifiers in column "Qual Valid Value List". List entries are found in tab "IM Valid Values".

How to implement: In Product 360 this kind of validation is ensured by the enumeration you add to a field.

Example: "organismCode" has valid value list "FNBOrganismCode" and is implemented as Enum.OrganismCode.WithOptionalCode at logical key ArticleMicrobiologics.LK.OrganismCode and field ArticleMicrobiologics.OrganismCode

Min. and max. values

Description: Numeric values, especially measurement values can be restricted to a certain range.

Where to find: Information can be found either in the Participant dictionary, in the columns "Length (All floats), Min. Length (All non-float Data Types)" and "Precision (All floats), Max. Length (All non-float)" or in the Validations document.

How to implement: In Product 360 this kind of validation is ensured by the entries in the field properties "Min. Range" and "Max. Range"

Example: "OrganismMaximumValue" has a max. length of 15 places

Other examples from the validations document:

- The value in - Qty of Next Level Item(s) (formerly Pack) is greater than 1 and less than 999999.
- If fatPercentageInDryMatter is not empty then value must be greater than or equal to 0 and less than or equal to 100.00.

Closely related to min. and max. values is the max. length of string values.

Example: GTIN Name: Value must be between 1 and 40 characters.

In this case use properties min. and max. length.

Mandatory fields

Description: Some fields are mandatory globally or mandatory in an optional or mandatory group.

Where to find: Information can either be found in the Participant dictionary, in column "Mandatory/Optional", or in the validations document.

How to implement: Set the lower bound to 1 in order to make a field mandatory within an entry of a sub entity.

Hint: Logical keys are always mandatory in Product 360.

Examples: GTINName is a required field

More information on configuration of the repository can be found in the section "[Data model\(see page 235\)](#)"

You should think about these kind of validations now!

Data Quality

Then there are more complex validations that are affecting multiple fields at once, depend on a specific value or target market. Most likely they will be implemented using DQ rule configurations.

Examples:

- If promotionalTypeCode is populated, then isConsumerUnit must be true.
- For each occurrence of the Loopgroup "promotional", attributes freeQtyOfNextLowerLevel and freeQtyOfProduct cannot both be populated.
- If targetMarketCountryCode is equal to '752' then packagingMaterialTypeCode and packagingMaterialCompositionQuantity are used in pairs. I.e. if one is populated the other one must be populated, too.
- If grossWeight and netWeight are provided on the same record, grossWeight must be greater than or equal to netWeight
- There must be at most one iteration of minimumFishMeatPoultryContent per Unit Of Measure

For further information see chapter "[Data validations\(see page 265\)](#)"

Export

When you have to output data into export files you should ensure to create well-formatted values, details can be found in the chapter "[Data validations](#)(see page 265)".

Summary



Analyze Module Summary

1. Get your documentation documents
2. Collect the fields of your module
3. Get an idea of the structure intended by GDSN
4. Try to fit the structure in an existing entity type
5. Find your logical keys
6. Implement the entity with the information from the section "Data Model"
7. Think about validations
8. In the process note all assumptions, discrepancies and open questions for later testing.

2.1.9.4 Data model

Introduction

This chapter describes how to implement a module as an entity. However, references to the additional documentation about the repository are made and can be found in the chapter "Domain Model (Repository)" of the Product 360 Documentation.

After you analyzed the new module as described in the chapter "[Analyze requirements](#)(see page 222)" of this documentation you have an idea what the entity will look like. This site describes in more detail the technical implementation and configuration possibilities. There are the following sections

- Create entity (Types area, custom area and sub entities)
- Create logical keys
- Create fields
- Create enumerations (valid value lists)
- Test checklist

If you only need to add a specific field, you can skip the first two sections.

Resources

In the following chapters we use these files:

- Profile Overview: FMCG_DIY_ARGO_ProfileOverview_Codelists_<Version>.xlsx
- Participant Dictionary: IM_Participant_Dictionary_R<Version>.xlsx
- Product 360 Documentation: Informatica MDM - Product 360 - v<Version> - Knowledgebase, Installation and Customization.zip

Create a new entity

Create a new entity - Types area

In the types area, there are three GDSN related sub entity types of `ArticleType`

- `ArticleDomainType`
- `ArticleMarketExtensionType`
- `ArticleDomainExtensionType`

ArticleDomainType

If you create a new module, this entity type should be your first choice when searching for an entity type to build your entity on.

It contains lots of field types of various data types. There are two main types:

- `ArticleDomainType<Subentity>.Std_<datatype>_<fieldNumber>`
- `ArticleDomainType<Subentity>.Res_<datatype>_<fieldNumber>`

The existing fields for GDSN modules in the standard are using the `ArticleDomainType<Subentity>.Std_<datatype>_<fieldNumber>` fields for GDSN attributes. Whereas the `ArticleDomainType<Subentity>.Res_<datatype>_<fieldNumber>` can be used for enriching those GDSN modules by customizations.

If you create an entirely new module, you are allowed to use `ArticleDomainType<Subentity>.Std_<datatype>_<fieldNumber>`. If you enhance an existing module, please only use the `ArticleDomainType<Subentity>.Res_<datatype>_<fieldNumber>` fields.

There is a sub entity called `ArticleIngredientLangType`. Don't use it. It is a special purpose sub entity only used by the standard.

If your running out of fields, you either create another entity or file a request via Product360 Support for the necessary changes.

ArticleMarketExtensionType

In contrast to the `ArticleDomainType` the `ArticleMarketExtensionType` has a different set of logical keys. It has a `PartyProxy` logical key, but misses the two `<entity>.LK.Std_LK_<datatype>_<keyNumber>` logical keys. It depends on your module which entity type to choose.

ArticleDomainExtensionType

Don't use this entity type. It is deprecated.



Use `ArticleDomainType` for new Modules.

Use `ArticleDomainType<Subentity>.Res_<datatype>_<fieldNumber>` to enhance existing modules.

Use `ArticleDomainType<Subentity>.Std_<datatype>_<fieldNumber>` fields in new modules, preferably.

Available ArticleDomainType sub entity types

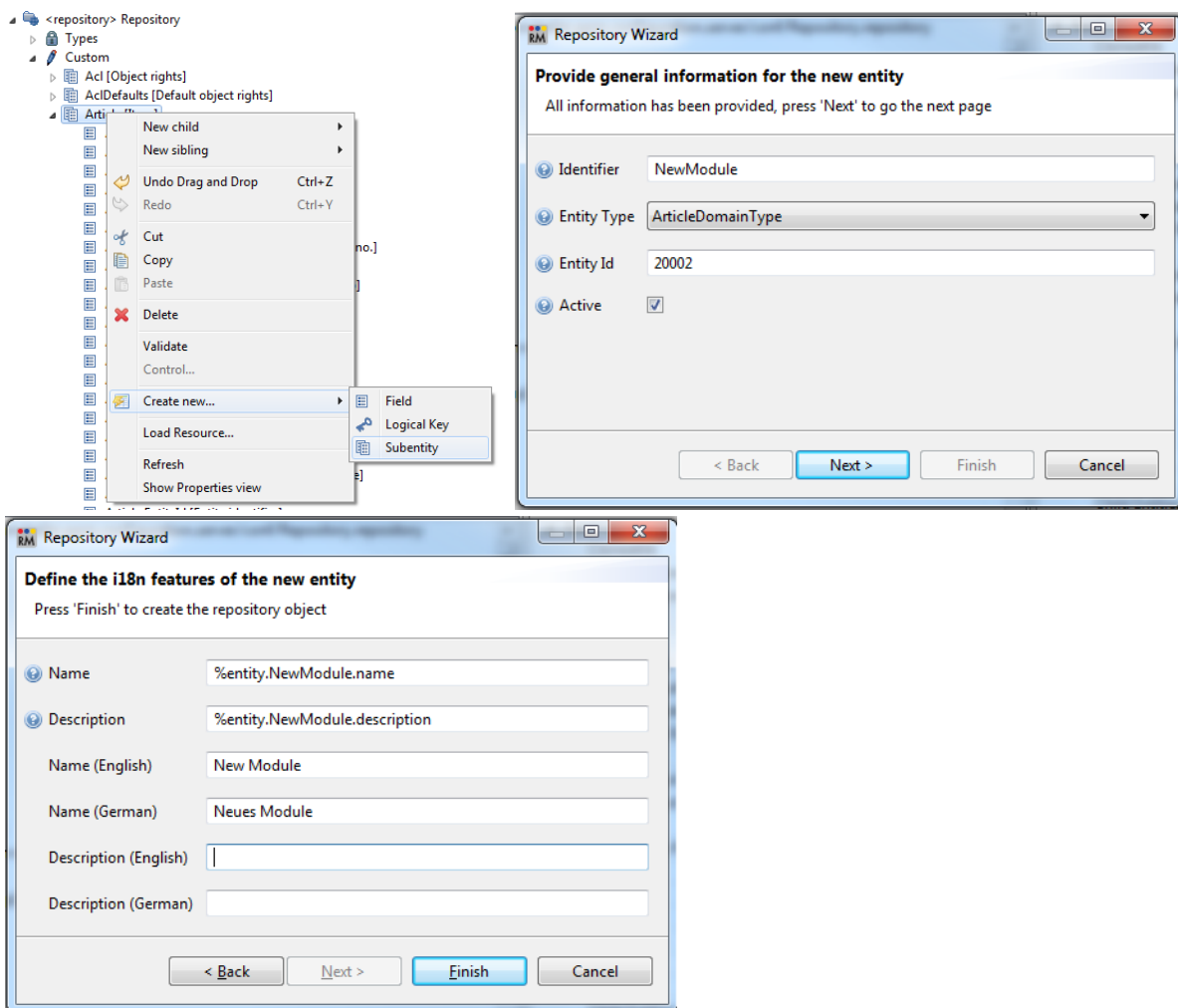
ArticleDomainType has several sub entity types commonly used in GDSN for different purposes

- **ArticleDomainLangType**
Entities based on this entity type contain language specific data. It is similar to ArticleLang and has logical keys EntityId and Language.
- **ArticleDomainUOMType**
Entities based on this entity type usually contain measurement values - a numeric value combined with a unit. The key UOMType can have the one of the values "metric" and "imperial". It is based on the assumption that different metric units like gram and kilogram can be converted in each other and therefore it would be an error source to store values for both units. But it may not be possible to automatically convert from metric units into imperial units.
Example: ArticlePackagingUOM.PackagingWeight and ArticlePackagingUOM.PackagingWeightUOM
- **ArticleDomainPartyType**
Entities based on this entity type usually contain data related to a party like customer specific data.
- **ArticleSubDomainType**
Entities based on this entity type usually contains data that together forms a list.
There are subentities supporting language specific data and measurement values for each list entry.
Example: ArticlePackagingMaterial contains a list of different packaging materials. A bottle of wine may be packaged in a hardwood box, padded with hemp fibers and surrounded by a cardboard box.

Create a new entity - Custom area

Wizard

Included in the Repository Manager are wizards helping you to create sub entities, fields, logical keys etc.



Create the entity of your module

There is nothing special to a GDSN module entity. It should support standard functionality like:

- Import
- Export
- Merge
- Clone
- Search
- Data Quality
- Service API



Entity IDs must be 20002 or higher.

Name and Description have to be externalized and be available in your needed client languages. Please note that the default language is English.

Create Logical keys

This is the wizard to create new logical keys:

The image displays three sequential screenshots of the 'Logical Key Wizard' dialog box within the 'Repository Wizard' application.

First Screenshot: The 'Logical Key Wizard' window shows the initial configuration. The 'Identifier' field contains 'ExampleEntity.LK.ExampleKey'. The 'Logical Key Type' dropdown is set to 'ArticleDomainType.LK.Channel'. The 'Value' field is empty. The 'Editable' checkbox is checked. Navigation buttons at the bottom include '< Back', 'Next >', 'Finish', and 'Cancel'.

Second Screenshot: This screen shows the 'Supports import' checkbox checked. Below it are two empty dropdown menus for 'Enumeration' and 'Proposal Enum'. The same navigation buttons are present at the bottom.

Third Screenshot: This screen displays a validation error: 'Name (English) is mandatory' with a red 'X' icon. The form fields are populated with placeholder text: 'Name' is '%logical-key.ExampleEntity.LK.ExampleKey.name', 'Description' is '%logical-key.ExampleEntity.LK.ExampleKey.description', and 'Name (English)', 'Name (German)', 'Description (English)', and 'Description (German)' are all empty. The 'Next >' button is disabled, while '< Back', 'Finish', and 'Cancel' are active.

In the new entity, there need to be a logical key for each logical key type of the underlying entity type. If you don't want to use all of them all you can deactivate these logical keys. Don't forget to add a default value in this case.

Each logical key needs a corresponding field and if you want to restrict the values for the logical key to a defined set using an enumeration, don't forget to add the enumeration to the logical key as well as to the field.



- Create a logical key for each logical key type in the underlying entity type.
- Each logical key needs a corresponding field.

If you want a logical key to be *active*, check this:

- Purpose is set to 1
- Editable is set to true
- It has an identifier
- Supports import is set to true

If you want a logical key to be *inactive*, check this:

- Purpose is set to 0
- Editable is set to false
- It has a value (default value)
- supports import is set to false

Logical keys - Service API

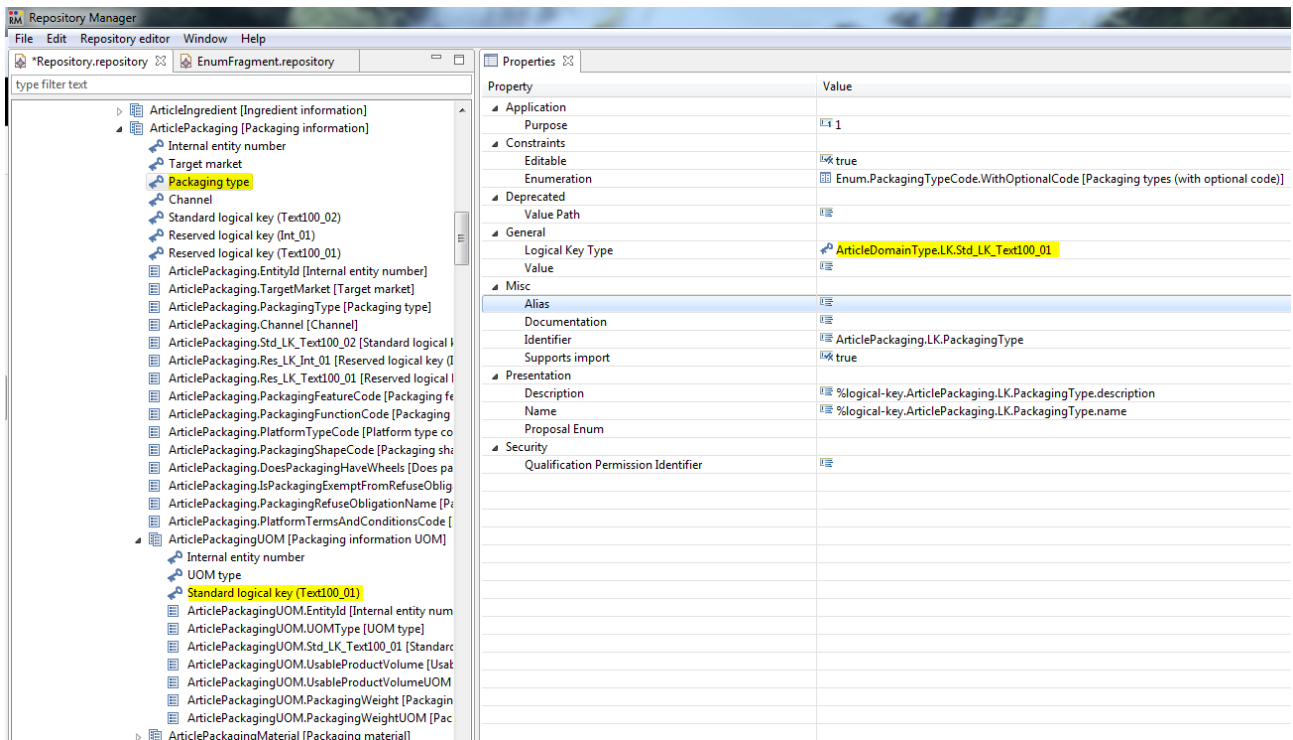
In the Service API, if you want to read or write a certain value, you need to specify the qualification. The qualification consists of values for all the logical keys on the path to the target field. To identify a logical key, the alias is used. You can define the alias in the selected line in the screenshot of the Repository Manager below. The alias gives you the opportunity to make the Service API requests more easily readable.

If there is no alias, the object name of the mapped field of the logical key is used. In the `ArticleDomainType`, the object names are generic because this entity type is the basis for a couple of different entities. Therefore it is recommended to use the alias.



The alias (or object name respectively) needs to be unique in the path to the target field.

Example: Let's have a look at the field `ArticlePackagingUOM.PackagingWeight`. Let's further assume a customer is using the Standard logical key (`Text100_01`) (set editable = true) in the entity `ArticlePackagingUOM`. Note that there is a logical key `Packaging` type that is based on the Standard logical key (`Text100_01`) in the entity `ArticlePackaging`.



The corresponding fields of both logical keys have the object name "std_LK_Text100_01".

You will notice that you get the values via Service API but not the complete qualification. In the code block below the packaging type is missing.

GET <http://localhost:1512/rest/V1.0/list/Article/ArticlePackaging/byCatalog?fields=ArticlePackagingUOM.PackagingWeight&catalog=MASTER>

```
{
  "cacheId": "20170329_111826_0",
  "entityIdentifier": "ArticlePackaging",
  "totalSize": 66,
  "startIndex": 0,
  "pageSize": 100,
  "rowCount": 2,
  "columnCount": 0,
  "columns": [],
  "rows": [
    {
      "object": {
        "id": "224435@1",
        "label": "Item1",
        "entityId": 1000
      },
      "qualification": {
        "targetMarket": "Barbados",
        "uomType": "metric",
```

```

        "std_LK_Text100_01": "DEFAULT"
    },
    "values": [
        "9.9"
    ]
},
{
    "object": {
        "id": "224435@1",
        "label": "Item1",
        "entityId": 1000
    },
    "qualification": {
        "targetMarket": "Barbados",
        "uomType": "metric",
        "std_LK_Text100_01": "DEFAULT"
    },
    "values": [
        "11.11"
    ]
}
]
}

```

If you try to write values and you specify values for all logical keys as in the example below

POST <http://localhost:1512/rest/V1.0/list/Article/ArticlePackaging>

```

{
    "columns": [
        {
            "identifier": "ArticlePackagingUOM.PackagingWeight"
        }
    ],
    "rows": [
        {
            "object": {
                "id": "'Item1'@'MASTER'"
            },
            "qualification": {
                "targetMarket": "Barbados",
                "std_LK_Text100_01": "Ampoule",
                "uomType": "METRIC",
                "std_LK_Text100_01": "DEFAULT"
            },
            "values": [
                "8.8"
            ]
        }
    ]
}

```

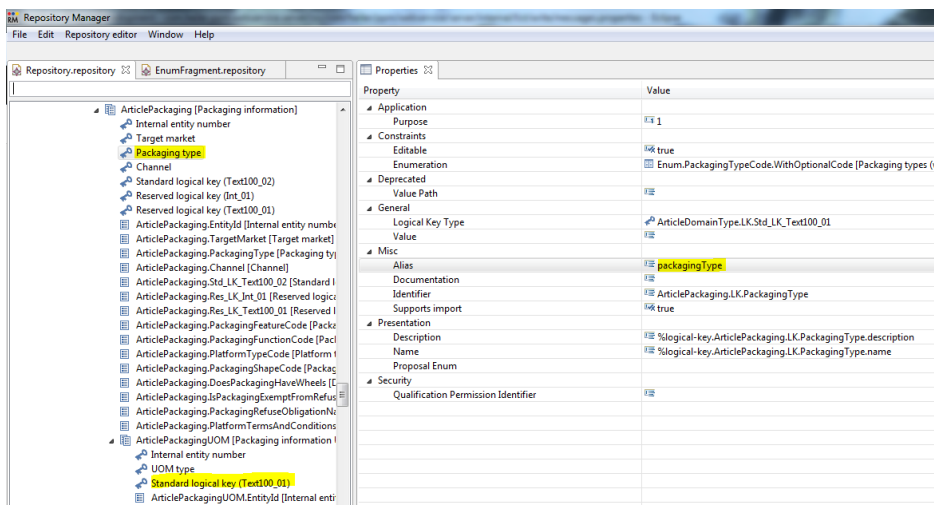
```
]
}
```

You will get this error message:

Result

Found two identical logical key types std_LK_Text100_01 for the same entity hierarchy. Please use the property 'alias' on the logical key in the repository to specify which logical key you want to use.

To fix this issue use a unique alias for all logical keys as recommended.



Don't forget to use the alias in your requests.

POST [http://localhost:1512/rest/V1.0/list/Article/ArticlePackaging \(with alias\)](http://localhost:1512/rest/V1.0/list/Article/ArticlePackaging (with alias))

```
{
  "columns": [
    {
      "identifier": "ArticlePackagingUOM.PackagingWeight"
    }
  ],
  "rows": [
    {
      "object": {
        "id": "'Item1'@'MASTER'"
      },
      "qualification": {
        "targetMarket": "Barbados",
        "packagingType": "Ampoule",

```

```

        "uomType": "METRIC",
        "std_LK_Text100_01": "DEFAULT"
    },
    "values": [
        "8.8"
    ]
}
]
}

```

Create a new field

Wizard

The Repository Manager also contains a wizard to create fields. A field has a lot of properties to configure. We won't show you all seven pages of the wizard at this point. Most important properties will be described below. More details can be found in the repository documentation mentioned above.

The left screenshot shows the 'Field Wizard' page with the following fields:

- Identifier: ArticleMicrobiologics.ExampleField
- Field Type: ArticleDomainType.Std_Bit_01
- Value:
- Active: ☒
- Editable: ☒

The right screenshot shows the 'Field Wizard' page with the following fields:

- Name: %field.ArticleMicrobiologics.ExampleField.name
- Description: %field.ArticleMicrobiologics.ExampleField.description
- Name From Top: %field.ArticleMicrobiologics.ExampleField.nameFromTop
- Documentation: %field.ArticleMicrobiologics.ExampleField.documentation
- Name (English):
- Name (German):
- Name from top (English):
- Name from top (German):
- Description (English):
- Description (German):
- Documentation (English):
- Documentation (German):

In the second screenshot you see the language specific properties of a field. The first four fields contain the property identifier for the externalization of the text. From the fifth field on you input the real labels you want to see in the UI.

Field Properties

Mandatory properties

First of all you need an **identifier** for your field. If there is no reason not to comply to the rule, this will be the same name as in the GDSN documentation. A Product 360 convention is that all fields of type Boolean start with the prefix "Is", e. g. "IsBaseUnit", "IsService", "IsConsumerUnit". As you can see in the examples, another convention is to use camel case for field identifier consisting of multiple words.

Next you have to choose a **field type**. Make sure the field type is not in use already. If you're using ArticleDomainType, you have to be careful with text field types. ArticleDomainType.Std_Text100_01 to ArticleDomainType.Std_Text100_10 have upper bound -1 meaning these are list types. Please always try to use a type field which is already configured as list/single value according to your needs. If you have a list of values, choose a string field type so that the whole list can be put in the single database

field. To choose the correct length for a field, multiply the length of a single value including a separator between the values with the number of expected values.

Example: If you have a field with an enumeration with four entries and the key of each enum entry is a single character, then it is safe to use a `Std_Text10_<number>` field.

Make sure the chosen field type is activated in the types area (`Inactive = false`).

If this field does not correspond to a non editable logical key, you should check that `Editable = true` is set.

Language specific properties

In order to see the field in the UI, it needs at least a **name**. You can define a **name from top** which is for example shown in the main table and normally includes the logical key values. For more information see section "Referencing on logical keys in field names" of the above mentioned documentation.

The **description** will be displayed as a tool tip in the field selection dialog and in the import. In general there is an English description in the GDSN documentation which should be used.

- Make sure the description is really a description of the data that should go in this field. Read the description before you add it to Product 360. Delete meaningless information like "0 to 80 character text field" or "Choose value from the drop down". This information is stored elsewhere in the repository.
- Use a meaningful description or leave it empty.
- If the description is a complete sentence, end it with a '.', if not don't use a '.

All labels have to be externalized in your client languages. For more information see section "Multi-language support of the repository" in the above mentioned documentation.

There is also a field **documentation** in the repository. The value of this field does not appear in the UI and does not need to be internationalized. It can be used for any documentation purpose and will be visible in the Repository Manager.

Support different functionalities

There are some properties defining if a certain field supports a specific functionality which are self explaining like "Cloneable", "Mergeable", "Searchable" and "Supports Data Quality".

Export

If the Export purpose is set to '0' the field cannot be exported. The export supports different purposes as described in the section "Export Purpose" of the above mentioned documentation. If you are not sure, start with "Export Purpose" = 1.

Import

If the Import purpose is set to '0' the field cannot be imported. If the field should be importable, set the "Import Purpose" = 1. For historic reasons make sure that the deprecated property "Purpose" always has the same value as the Import Purpose and that supports import is set accordingly.

To determine where this field should be displayed in the "Repository" tree view of the import perspective use the property "Category". That's why it's recommended to keep all fields of one GDSN module in the same category. However, if you have reasons to do it differently, it is possible. The category itself also needs to be created in the repository.

Service API

If the field should be available in the Service API, you need to set "Supports service API" = true. For technical reasons the field has to support the service API if you want to use Data Quality (Supports Data Quality = true).

Constraints

Occurrences: The properties "Upper Bound" and "Lower Bound" define how many values can be stored here.

- For a mandatory value set both properties to 1.
- For a non-mandatory field with one value set lower bound = 0 and upper bound = 1.
- For a list of values set lower bound = 0 and upper bound = -1.

You can define if it should be an INFO, a WARNING or an ERROR in case the upper or lower bound is not met. Normally this should be set to ERROR, otherwise the system will store these values regardless of the field configuration.

Valid value lists: There are two places where you can add an enumeration to the field.

- Enumeration - The value of this field must be one of this enumeration. Otherwise the user will get an error and the value won't be stored. If the database already contains values not contained in the enumeration these will no longer be displayed in the UI.
- Proposal Enum - Use the proposal list if the user is allowed to store values which are not contained in the list.

Since the enumeration keys are stored as the values for this field, the chosen field type has to have a matching data type with the key class of the enumeration.

Field length: GDSN often defines the max. field length.

- For string values use the properties "Min Length" and "Max Length". Make sure the values you choose here are not lower resp. higher than the min. and max. length in the types area for the corresponding field type. If the field contains a list of values (upper bound = -1), the min. and max. length refer to a single entry, not the entire list.
- For numeric values the "Min Length" and "Max Length" are not used. If you want to restrict a numeric value you have to use "Range Min" and "Range Max". Make sure the values are not lower resp. higher than the min. and max. range defined for the corresponding field type. The decimal separator used here is '.'.
You can define how many decimal places should be displayed in the UI by using the property "Scale". The displayed value will be rounded. Note that this is only a matter of presentation. In the database all decimal places given in the UI are stored.
To ensure that only values in the defined range will be stored, set "Validation Severity Range" to ERROR. The default value is WARNING.
- For dates also use "Range Min" and "Range Max". This is an example of the pattern you have to use: "yyyy-MM-dd HH:mm:ss". For example: 9999-12-31 23:59:59

If you don't restrict the values in the custom area the default values from the corresponding field type will be used.

Transitions: If you store a proxy in the field and you want to make fields of the proxy available (for example in the field selection dialog) you can set the matching root entity to the property "Proxy Transition Entity". More information about transition fields can be found in the knowledge base article at chapter "Transition fields - Group name" of the Product 360 Documentation.

Presentation

"Visible" and "Visible From Top":

- Your field belongs to a deactivated logical key: Set "Visible" and "Visible From Top" to false. The field should not appear in the UI.

- Your field belongs to an activated logical key: Set "Visible" to true, so it will be visible in the sub entity views. Set "Visible From Top" to false since we don't want fields which have to be qualified with the value they contain in the main table.
- Your field doesn't belong to a logical key: By default set "Visible" and "Visible From Top" to true. The field should be available in the main table as well as in the sub entity views.

In order to have a display name in the UI, don't forget to fill in the property "Name" if "Visible" is set to true and property "Name From Top" if "Visible From Top" is true.

View configuration:

You can configure that a field is present in the default configuration of the corresponding sub entity view. Set "Display By Default" = true. This configuration will be overwritten by the layout stored in the client's workspace. You can also configure in which order the fields should appear in the table using "Default Column Order". The index is 0 based.

You can do the same for the main table using "Display By Default From Top" and "Display Column Order From Top". However this is not recommended for GDSN modules.

Miscellaneous

Default value: In case you create a field belonging to a logical key that is not editable, not only the logical key needs a default value but the field needs the same default value. The default value may also make sense for other fields, especially for mandatory fields.

Limitations

- In some cases the Item Management data model allows values from 0 - 9,999,999,999 for attributes with datatype "Integer". Due to technical restrictions Product 360 allows only values from 0 to 2,147,483,647 for attributes with datatype "Integer". In real life this should not cause any issues because it's unlikely to have such a large value for any attribute.
- IM defines decimal values in a flexible way by defining the complete length and the max. number of decimal places.

Example: complete length 15 - max. decimal places 15

- 1234567890,12345 is valid
- 12345,1234567890 is valid
- 12345678,12345678 is not valid because the complete length is greater than 15

Product 360 does not support this flexible definition. There is a technical limitation to 10 places before the decimal separator and 6 decimal places. Higher numbers are not supported.

- There is no way to limit the decimal places that are stored in the database. The "Scale" value is only used to format a value in the UI.

Deactivate a GDSN entity

If you want to deactivate a GDSN entity read the chapter "Hide a GDSN module" in the "[Repository configurations\(see page 64\)](#)" section of the GDSN Accelerator documentation.

Create or adjust a valid value list

Many fields need a valid value list. In most cases this is a simple list of some kind of codes mapped to language-dependent labels. Sometimes you need a list of units of measure. First of all, you should check if the needed valid value list is already available.



Synonyms for "valid value list" are "look up values", "preset values" and in technical terms "enumeration".

Generate or adjust a repository enumeration

Repository.repository vs. EnumFragment.repository

The "Repository.repository" file contains all enumerations used by repository fields. For GDSN or food and beverage enumerations the repository only contains the definition of the enumeration. The actual enumeration entries are contained in a second file called "EnumFragment.repository". This will keep the repository more organized because there are some GDSN valid value lists with high numbers of entries.

The screenshot displays the Repository Manager application interface. The top window shows the 'Repository.repository' file, which lists various enumeration types such as 'Enum.DictionaryTypes', 'Enum.DietTypeCode', 'Enum.DietTypeSubcode', etc. The bottom window shows the 'EnumFragment.repository' file, which contains a detailed list of enumeration entries under the 'Enum.DietTypeCode' category, including 'Coeliac (COELIAC)', 'Dietetic (DIETETIC)', 'Free from gluten (FREE_FROM_GLUTEN)', 'Halal (HALAL)', 'Kosher (KOSHER)', 'Non-vegetarian - GSI SA use only (NON_VEG)', 'Vegetarian-honey - GSI SA use only (VEG_HONEY)', 'Vegetarian-lacto (milk) - GSI SA use only (VEG_LACTO)', 'Vegetarian-ovo (egg) - GSI SA use only (VEG_OVO)', 'Vegan (VEGAN)', 'Vegetarian (VEGETARIAN)', 'Without beef (WITHOUT_BEEF)', and 'Without pork (WITHOUT_PORK)'. The Properties panel on the right shows details for the selected enumeration, including 'Active' (true), 'Class Name' (com.heiler.ppm.repository.enumerations.StdEnumProvider), 'Identifier' (Enum.DietTypeCode), 'Key Class Name' (java.lang.String), 'Case Sensitive' (false), 'Documentation' (Enum Entries are defined in EnumFragment.repository), 'Service alias', 'Presentation' (Description: %enum.DietTypeCode.description, Name: %enum.DietTypeCode.name), and 'Security' (User Specific: false).

Create enumeration

Use the Repository Wizard of the Repository Manager to create a new enumeration.

The image displays three sequential screenshots of the 'Repository Wizard' dialog box, specifically the 'Enumeration Wizard' tab, used for configuring a new enumeration.

First Screenshot: The 'Enumeration Wizard' dialog box is shown with the title 'Configure the general options of an enumeration'. It contains four fields: 'Identifier' (set to 'Enum.'), 'Class Name' (set to 'com.heiler.ppm.repository.enumerations.StdEnumProvider'), 'Key Class Name' (set to 'java.lang.String'), and 'Active' (checked). Navigation buttons at the bottom include '< Back', 'Next >', 'Finish', and 'Cancel'.

Second Screenshot: The dialog box shows the message 'All information has been provided, press 'Next' to go the next page'. It contains two unchecked checkboxes: 'Case Sensitive' and 'User Specific'. Navigation buttons at the bottom include '< Back', 'Next >', 'Finish', and 'Cancel'.

Third Screenshot: The dialog box shows a validation error: 'Name (English) is mandatory'. It contains fields for 'Name' (set to '%enum.ExampleEnum.name'), 'Description' (set to '%enum.ExampleEnum.description'), and three additional fields for localization: 'Name (English)', 'Name (German)', 'Description (English)', and 'Description (German)'. Navigation buttons at the bottom include '< Back', 'Next >', 'Finish', and 'Cancel'.

Use an identifier that makes it easy to find the corresponding field.

Usually you need to use the key class "string" when creating an enumeration for GDSN.

Enter the properties identifier in the fields "Name" and "Description", shown in the third screen. They should have the pattern %enum.<identifier without substring 'Enum.'>.<name|description> as can be seen in the screenshot above. There is no "Name from top", but name and description should be provided at least in English. In case you are using more client languages, please maintain those names in the according "repository.properties" file for your language.

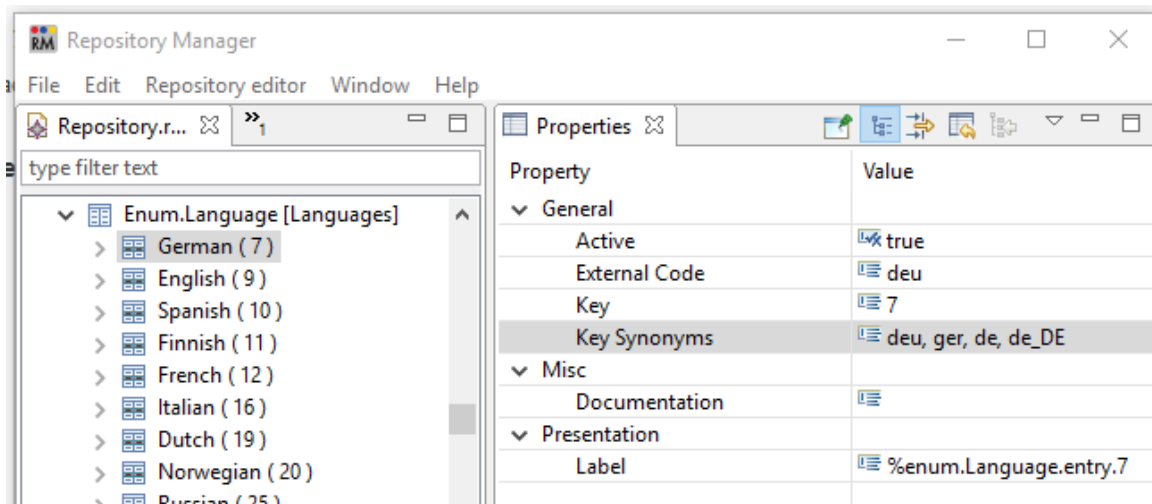
Copy the enumeration into "EnumFragment.repository" in order to add entries in the next step.

For a good trace-ability of your enumeration definition, it is recommended to add a documentation to your enumeration in the "Repository.repository" file: "Enumeration entries are defined in EnumFragment.repository".

Add enumeration entries

Key synonyms come in handy if there are multiple different values in the data source which should result in the same value in Product 360.

Example: Enum.Language contains enum entries with synonyms. For example you can import "deu", "ger", "de" or "de_DE" and either value will result in the language key 7 in the system.



Make sure the label is externalized and available in English and your required client languages.

Delete enumeration entries

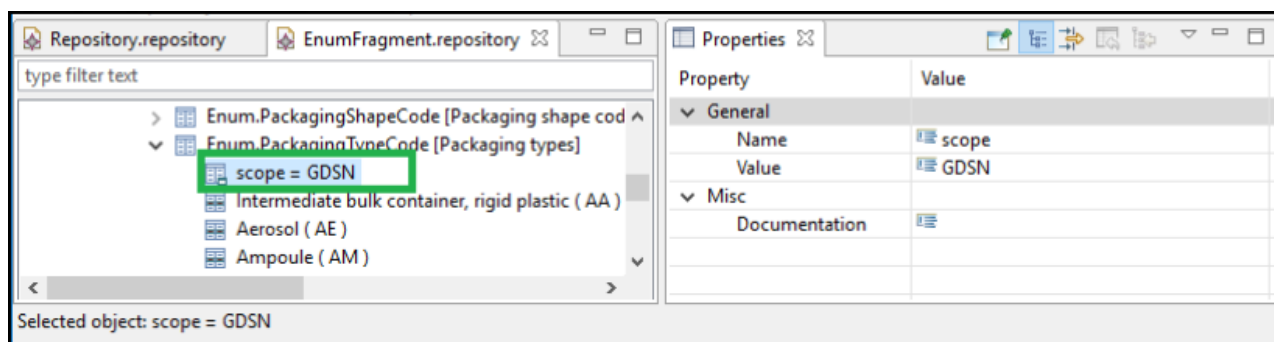
In a customizing, it is allowed to delete enumeration entries.

Example: If your customer is a manufacturer of vegan food, you may want to delete the enumeration entry "MEAT" in Enum.DietTypeSubcode in order to reduce false data entered by accident.

Be aware that if the keys of the entries you delete are already in the database or could be imported from another system, these values won't be displayed anymore.

Scopes on enumerations and enumeration entries

You may have seen parameters at enumerations or enumeration entries named "scope".

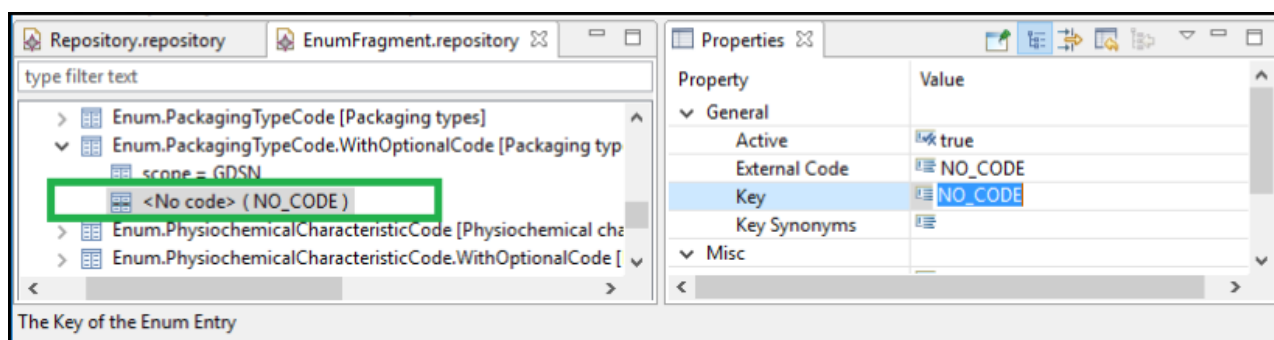
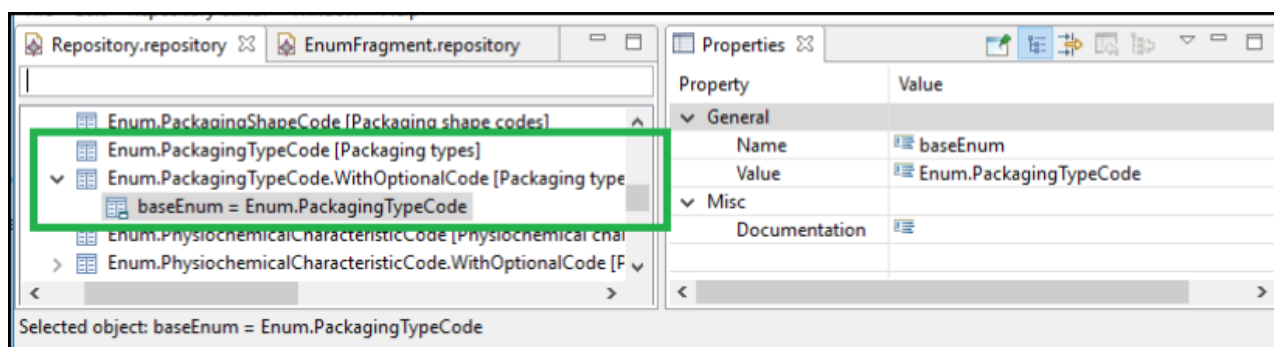


You should not change it or create such parameters at your enumerations or enumeration entries. Those parameters are needed for internal purposes only; they are used to make automatic repository adjustments during startup according to the configuration defined in "application_modules.properties" file.

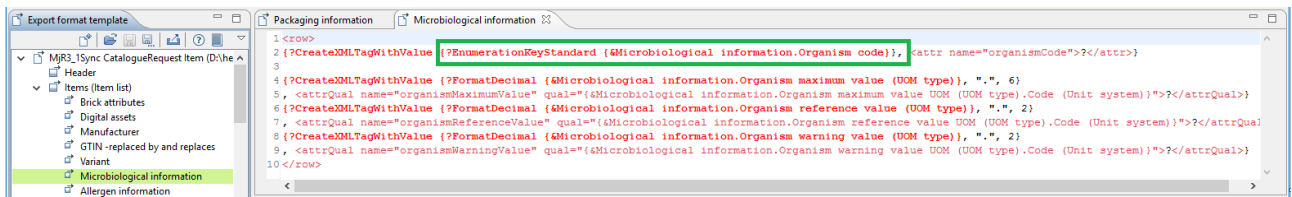
Create enumerations with optional code

If you have to enable a mandatory field with a valid value list to get "no value" for an entry (such fields are usually logical key fields), you have to use an enumeration with optional code(s). Such enumerations provide all values of the corresponding valid value list and one or more values that will not be transferred to the GDSN pool.

First you create an enumeration as described above, the empty enumeration definition in "Repository.repository" and the entries in "EnumFragment.repository". Then you create a second enumeration with an identifier like "<Identifier of first enumeration>.WithOptionalCode". That second enumeration gets an entry standing for "no code". All other enumeration entries will be used from the "parent" enumeration.



If you have to adjust the corresponding export template to transfer item data to the GDSN pool you should use the EnumerationKeyStandard export function. Details on that function can be found in the ["Technical details\(see page 84\)"](#) chapter of this GDSN documentation.



Create or adjust a unit of measure list

The GDSN unit system contains many units of measure. Usually, a valid value list for a field only uses some of those units.

Therefore units are separated into unit categories but unfortunately the categories are different depending on the GDSN pool which is used.

Product 360 contains predefined categories for the IM GDSN system units. For DSE the same categories can be used or new categories can be created as described in the chapters below.

To see a list of valid units for a field open the appropriate GDSN document. Use the Participant Dictionary when using the IM GDSN pool or the Profile Overview when using the DSE GDSN pool.

GDSN unit categories

The GDSN units are separated into categories. One unit can be in different categories and one field can have valid values of multiple categories.

A list of units and their category can be seen in the *Units* perspective of Product 360. Select the GDSN unit system and add the field "Category" with the field selection dialog. Make sure to qualify the "Category" field with the GDSN parameter.

All units in the unit system "Own GDSN"			
	Code (Own GDSN)	Name (Own GDSN, English)	Category (Own GDSN)
1	BFT	board foot	Volume units
2	EA	each	Count units
3	BP	hundred board feet	Volume units
4	HC	hundred count	Count units
5	CWA	hundred pounds (cwt)/hun...	Mass units
6	MIU	million international unit (...)	Count units
7	LTN	ton (UK) or long ton (US)	Mass units
8	STN	ton (US) or short ton (UK/US)	Mass units

Use an existing unit enumeration

In order to use a category at a specific field it is necessary to use an unit enumeration. Product 360 has predefined enumerations named after the categories for the IM data pool. In order to map a unit enumeration to a specific field it is best practice to use the Repository Manager.

Property	Value
▼ Application	
Cloneable	true
Export Purpose	1
Import Purpose	1
Mergeable	true
Purpose	1
Searchable	true
▼ Constraints	
Editable	true
Enumeration	Enum.GDSNAreaUnits [GDSN Area units] ▼
Lower Bound	Enum.GDSNAreaUnits [GDSN Area units]
Max Length	Enum.GDSNConfirmationMessageCode [GDSN confirmation message code]
Min Length	Enum.GDSNCountUnits [GDSN Count units]
Picture Clause	Enum.GDSNDataRecipientMessageType [GDSN Data Recipient message type]
Proxy Transition Entity	Enum.GDSNDataSourceMessageType [GDSN Data Source message type]
Range Max	Enum.GDSNDataSourcePublicationMode [GDSN Publication mode]
Range Min	Enum.GDSNDataSourcePublicationRecipient [GDSN Publishate to]
Upper Bound	Enum.GDSNDataSourcePublishMessageType [GDSN Data Source publishing message type]
Validation Severity Enum	Enum.GDSNDensityUnits [GDSN Density units]
Validation Severity Lower Bound	Enum.GDSNDimensionUnits [GDSN Dimensions]
Validation Severity Range	Enum.GDSNEnergyUnits [GDSN Energy units]
Validation Severity Upper Bound	Enum.GDSNExecutedOperation [GDSN executed operation]
▼ General	
Active	true
Field Type	Enum.GDSNMessageInProgress [GDSN Message In Progress]
Identifier	Enum.GDSNMiscUnits [GDSN Miscellaneous units]
Value	Enum.GDSNNutritionQuantityUnits [GDSN Nutrition Quantity units]
▼ Misc	
Documentation	Enum.GDSNOrderUnits [GDSN Order units]
Supports Data Quality	Enum.GDSNPackagingUnits [GDSN Packaging units]
	Enum.GDSNPowerUnits [GDSN Power units]
	Enum.GDSNPressureUnits [GDSN Pressure units]
	Enum.GDSNProductYieldUnits [GDSN Product yield units]

Creating new unit enumerations

When the predefined unit enumerations do not meet the requirements of the field you want to add, it is possible to create a new enumeration in the repository by using the Repository Manager and include all categories which are needed separated by a ",".

It is important that the `unitSystem` property matches the unit system which should be used for the enumeration. The standard GDSN unit system is **70**.

- > Enum.GDSNProductYieldUnits [GDSN Product yield units]
- > Enum.GDSNProportionUnits [GDSN Proportion units]
- > Enum.GDSNPublicationMessageCode [GDSN publication message code]
- ▼ Enum.GDSNQuantityUnits [GDSN Quantity units]
 - UnitSystem = 70
 - UnitCategory = Area,Count,Dimensions,InfoStorage,Mass,Volume
 - Enum.GDSNResponseType [GDSN response type]

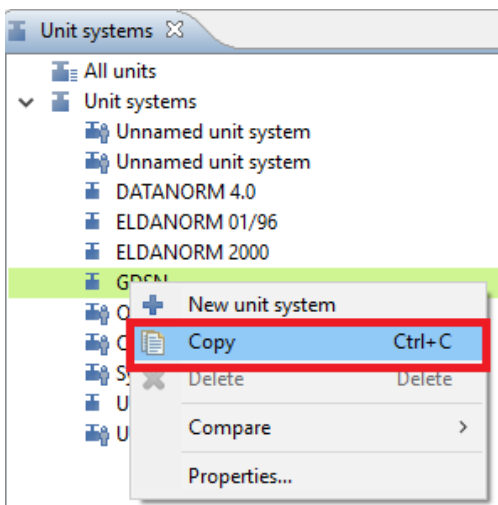
After creating the unit enumeration it can be referenced as enumeration at a field as shown above.

Creating a custom unit system



Beware when using a custom unit system. All changes made in the standard GDSN unit system are not automatically added to any custom unit system and have to be maintained manually!

When the predefined unit categories are insufficient or not usable for your project, it is possible to create a custom unit system based on the GDSN unit system. To do this simply copy the GDSN unit system in the *Units* perspective.



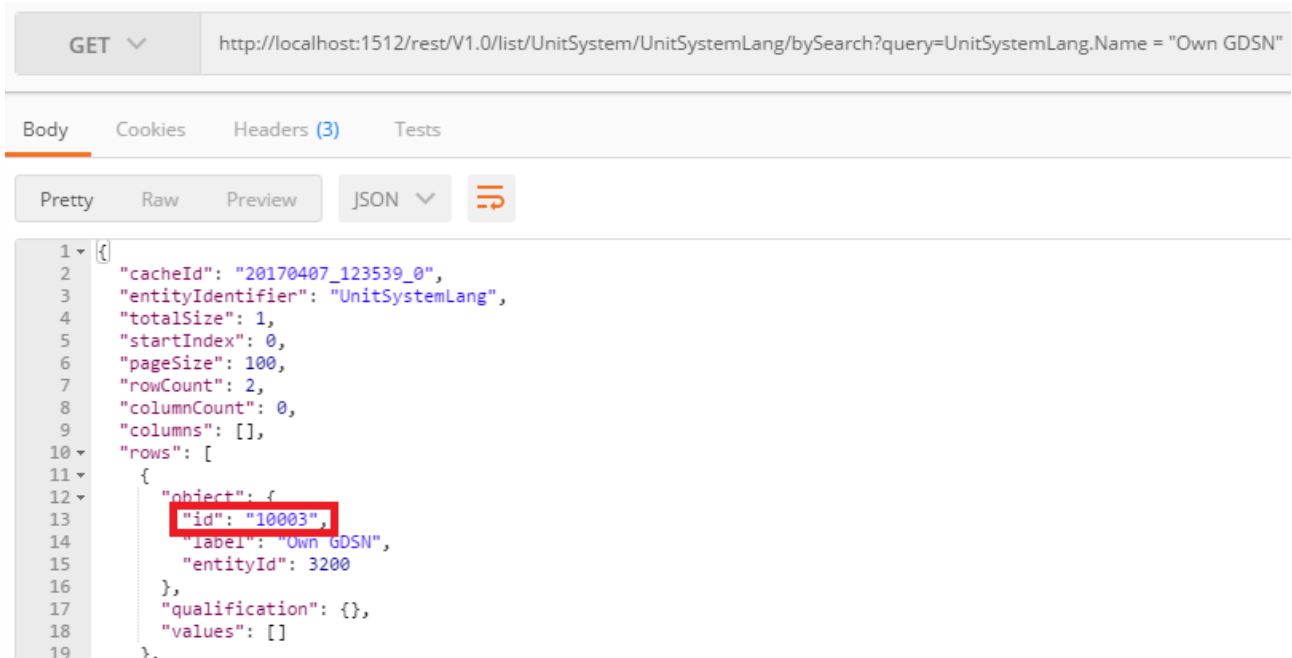
After that it is possible to add or modify units and their categories for your needs in the *Unit maintenance* view.

To use the unit system created this way some additional steps have to be done in order to use it in the whole application:

Get the custom unit system ID

As stated above GDSN has a default unit system and in order to change that the ID of the new unit system is needed.

To get the unit system ID the Service API can be used as shown in the picture below.

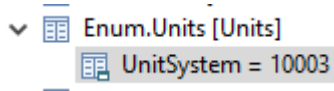


It is important to add a blank space before and after the "=" sign when qualifying UnitSystemLang.Name with a value.

The ID marked in red is the unit system ID of the custom unit system. To have Product 360 use the custom unit system instead of the GDSN unit system open the Repository Manager.

(Optional) Use the new unit system in all unit fields

To use the new unit system in all UOM fields, including the fields which are not GDSN related, navigate to the enumeration Enum.Units and change the parameter UnitSystem to the ID retrieved from the Service API.



When restarting Product 360 the custom unit system will be used instead of the standard unit system.

Create category enumerations

As described in the section "Creating new unit enumerations", it is needed to add unit enumerations that use the custom unit system as enumeration parameter.

Change export UOM fields to use custom unit system

Make sure to use the new unit system in the export for all UOM fields.

Create a new unit

If a unit is missing in the GDSN unit system and is available in the official GDSN documents please contact the support as they will provide a database script to add the unit.

Checklist: Test the module

Check now

- ☐ Start validation in repository editor (right-click on a node you want to be validated, choose "Validate")
 - ☐ Validate the new (sub) entity
 - ☐ Validate whole custom node
- ☐ Check data type, field length and valid values

This can be checked at different places. Try to write values of the correct and incorrect data type and field length or write correct and incorrect valid values using the import, the Service API or the UI. The positive check is not enough. If the import does not report an error when writing a valid value, it can either mean you

added the valid value list correctly or it can mean you forgot to attach the enumeration to your field. Always try to get the errors you expect.

- ☐ Check import
 - ☐ Check category
 - ☐ Check name (It's the name of the logical key, not the corresponding field, displayed in the import UI)
 - ☐ Check in the view "Field details" if the properties are as expected
- ☐ Check export
 - ☐ Only needed fields can be exported
 - ☐ No deactivated logical keys appear in the list of fields
 - ☐ Only needed entities are available as export sub-data types
 - ☐ The field names should contain all visible qualifications (logical keys)
- ☐ Check Service API
 - ☐ Look at the meta API
 - ☐ Invisible fields are not available
 - ☐ Names and descriptions are available
 - ☐ The valid value lists are correct
 - ☐ Read fields
 - ☐ Write fields
 - ☐ Check that each visible logical key has a unique alias within its path to the root entity

Check after you built the UI

- ☐ Are table or detail views available in the desktop client? Is detail tab in web client available?
- ☐ Check that all fields are visible
 - ☐ Check in main table (qualified access)
 - ☐ Check in detail table and detail tab (unqualified access)
- ☐ Check if data can be created, edited and be deleted in desktop client and web client
- ☐ Check datatype, field length and valid values (if not already done outside the UI)
- ☐ Check correct display names and descriptions
 - ☐ Are labels available in English and needed client languages?
 - ☐ Do names from top contain all necessary logical keys? Are the logical keys in the expected order?
You can check this either in the column headers or in the field selection dialog of the main table. It is recommended that the order of the logical keys is determined by the containing entity. Go from the root entity to the display fields and collect the visible logical keys on this path.
- ☐ Check if the item is still clone-able and merge-able after you maintained data
- ☐ Check if the item search works correctly
- ☐ Make sure the desktop view(s) contain(s) a suitable set of default columns and the layout of the web detail tab is suitable

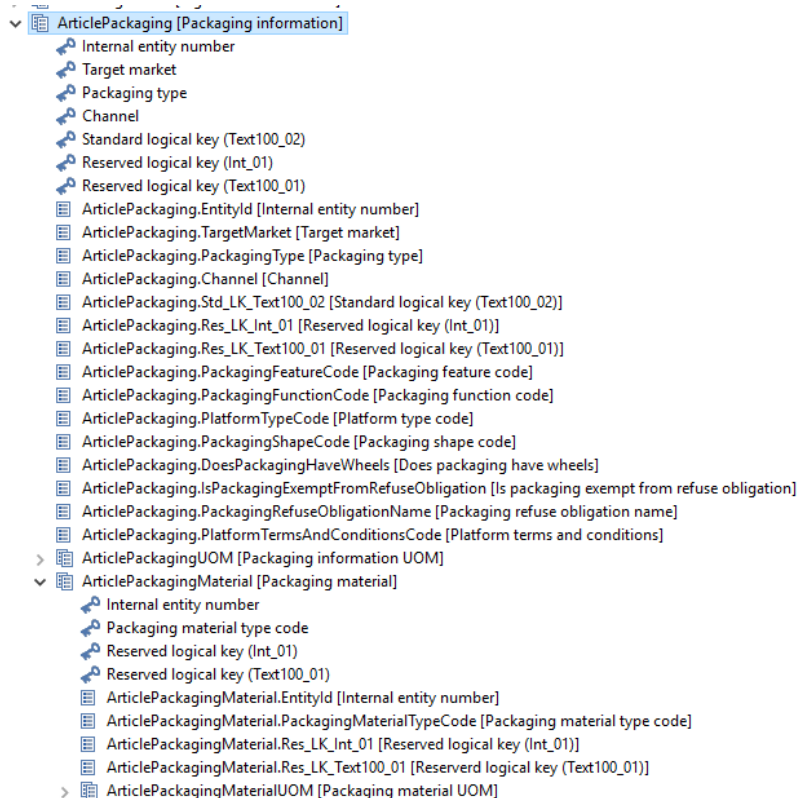
2.1.9.5 UI adjustments

- [Creating additional elements in Desktop UI](#)(see page 257)
 - [General information](#)(see page 257)
 - [Adding views](#)(see page 258)
 - [Third level hierarchy](#)(see page 259)
 - [Adding perspectives](#)(see page 259)
 - [Adding views to a perspective](#)(see page 260)
 - [Adding standard views to a custom perspective](#)(see page 260)
 - [Extend a standard GDSN perspective with custom views](#)(see page 261)
- [Creating additional elements in Web UI](#)(see page 261)
 - [General information](#)(see page 261)

- [Adding a new field](#)(see page 262)
- [Adding a new module](#)(see page 262)
- [Interface visibility and display rights](#)(see page 265)
- [Limitation](#)(see page 265)

Creating additional elements in Desktop UI

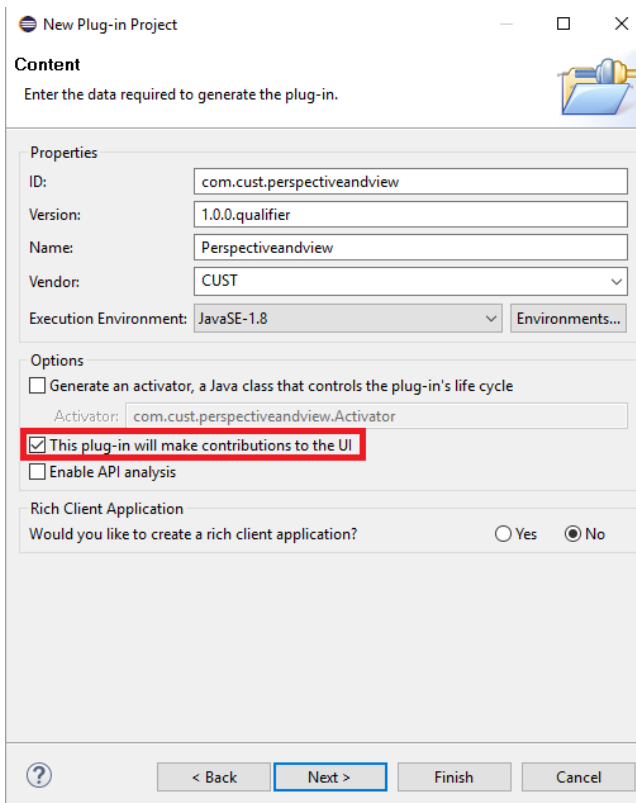
After we created our new entity `ArticlePackaging` we want to edit data using the UI of Product 360.



We will need a view for the sub entity `ArticlePackaging` and one for the sub entity `ArticlePackagingMaterial`. The customer decided that no additional views for the UOM or the Lang sub entities are needed as they can be seen in their appropriate higher level sub entity.

General information

To have all UI elements concentrated in a specific place it is best practice to create a separate plugin for the UI elements ending with `.ui`, for example `com.informatica.customizing.additionalviews.ui`. When creating the plugin please make sure that the checkbox "This pug-in will make contributions to the UI" is checked.



New Plug-in Project

Content
Enter the data required to generate the plug-in.

Properties

ID:

Version:

Name:

Vendor:

Execution Environment:

Options

☐ Generate an activator, a Java class that controls the plug-in's life cycle

Activator:

☒ This plug-in will make contributions to the UI

☐ Enable API analysis

Rich Client Application

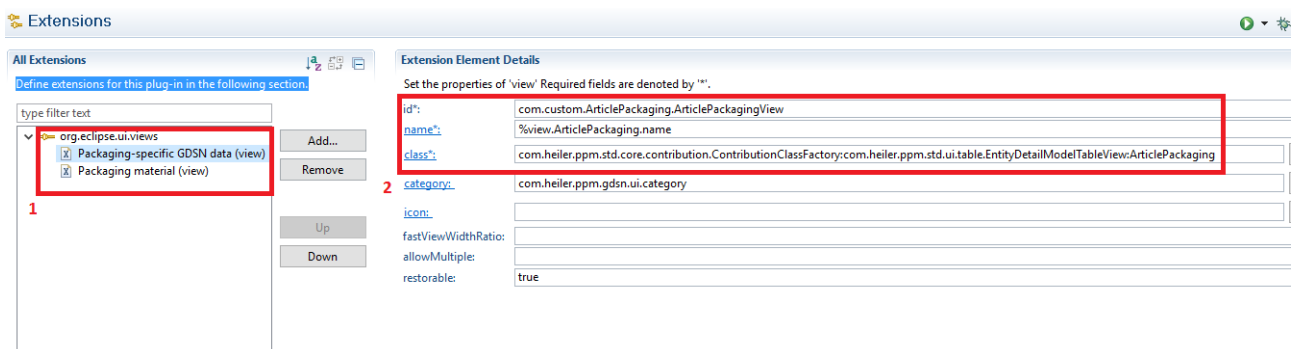
Would you like to create a rich client application? ☐ Yes ☒ No

When the box is checked a `plugin.xml` file will be created where we can add contributions to specific extension points.

After each headline the result can be seen by adding the new UI plugin to the plugin directory of Product 360 client.

Adding views

A view is used to show every field from the given entity. Every sub entity, besides UOM and Lang entities, needs a separate view.



Extensions

All Extensions
Define extensions for this plug-in in the following section.

type filter text

☒ org.eclipse.ui.views

☒ Packaging-specific GDSN data (view)

☒ Packaging material (view)

Extension Element Details

Set the properties of 'view' Required fields are denoted by '*'.

id*:

name*:

class*:

category:

icon:

fastViewWidthRatio:

allowMultiple:

restorable:

1. Contribute the views to the extension point `org.eclipse.ui.views`
2. Complete the mandatory properties marked with a *. Details about the properties can be found in the "Creating custom perspectives" section (chapter "Customizing") in the document "Informatica MDM - Product 360 - <VERSION>- Knowledgebase, Installation and Customization".

Because the `ArticlePackaging` entity is a direct sub entity of the root entity `Article` we can use the `com.std.ui.table.EntityDetailModelTableView` which is used by direct sub entities.

Sub entities which are not direct sub entities from a root entity are called third level hierarchy entities. They must use a different `TableView`.

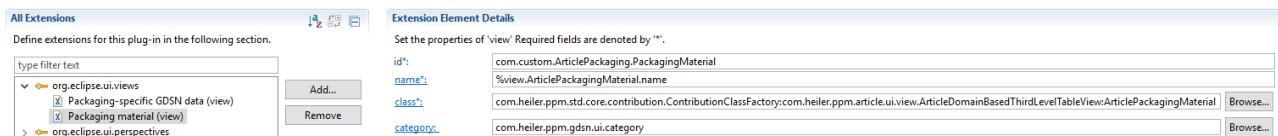
Third level hierarchy

In our example `ArticlePackagingMaterial` is a third level hierarchy entity (`Article` → `ArticlePackaging` → `ArticlePackagingMaterial`). When having a third level hierarchy in the repository it is necessary to use another value at the class text field because the content which is shown in the `ArticlePackagingMaterial` view should depend on the selection made in the `ArticlePackaging` view.

The `TableView` which we have to use

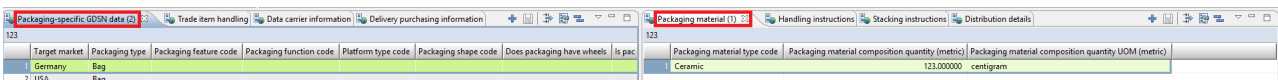
is: **`com.heiler.ppm.article.ui.view.ArticleDomainBasedThirdLevelTableView`**

So the configuration for the third level hierarchy view looks like this:



Take care when placing the ArticlePackagingMaterial view

It is necessary that both views can be visible at the same time and therefore we recommend to place the `ArticlePackaging` view on the opposite side. Otherwise the selection paradigm cannot work. This has to be done for every third level hierarchy view and their counterparts.



Adding perspectives

To add a perspective we first need to add a new java class to the UI plugin containing the new perspective with a perspective identifier.

GDSNCorePerspective

```

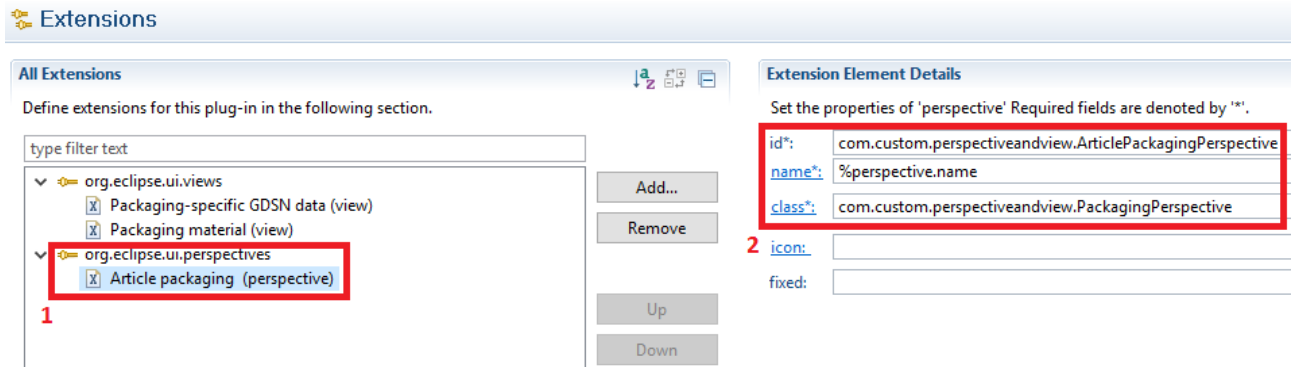
1  public class PackagingPerspective implements IPerspectiveFactory
2  {
3      public static String PERSPECTIVE_ID =
4          "com.custom.perspectiveandview.ArticlePackagingPerspective"; //$NON-NLS-1$
5
6      @Override
7      public void createInitialLayout( IPageLayout layout )
8      {
9          layout.setEditorAreaVisible( false );
10     }

```

```
10 }

```

Now we have to add a contribution to the `org.eclipse.ui.perspectives` extension point.



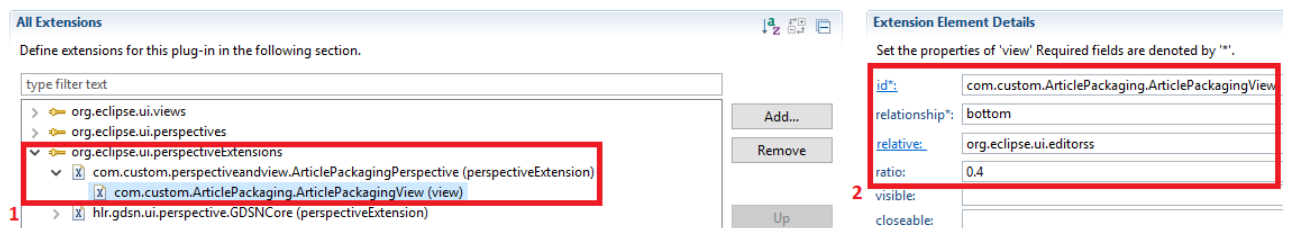
1. Add perspective to the extension point.
2. Fill the mandatory properties marked with a *. More information about perspective properties can be found in the "Creating custom perspectives" section (chapter "Customizing") in the document "Informatica MDM - Product 360 - <VERSION>- Knowledgebase, Installation and Customization".

In our case the value for the `class` attribute is the class we just created. The name has to be fully qualified with the package where the class is located.

Here: `com.custom.perspectiveandview.PackagingPerspective`.

The perspective can now be seen in the client but the perspective is empty. In the next chapter we will add views to the perspective.

Adding views to a perspective



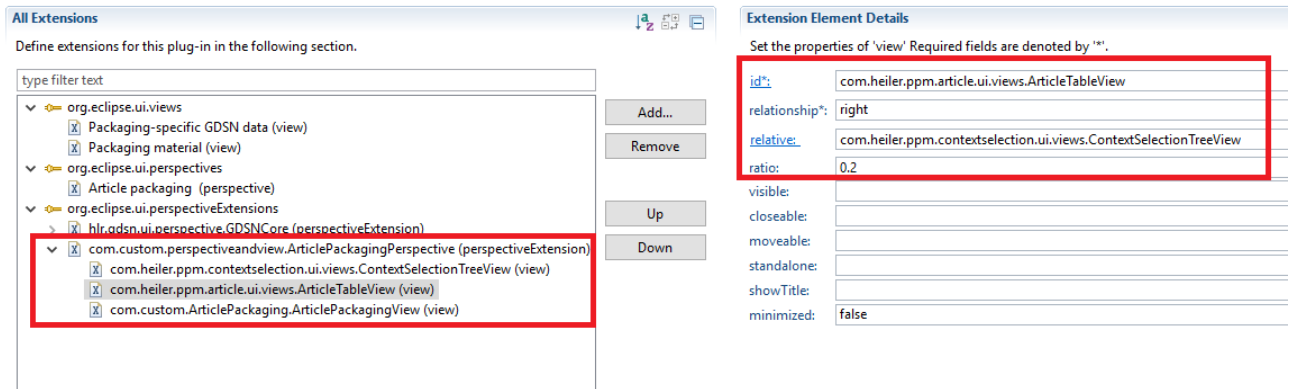
To add a view to an existing perspective you have to add a contribution to the `perspectiveExtensions` extension point.

1. The target `id` of the contribution `perspectiveExtension` has to be the `id` of the perspective where you want to make the modifications. In our example we want to add a view to our "ArticlePackagingPerspective" we created in the section above. So we use the `id` `com.custom.perspectiveandview.ArticlePackagingPerspective`. We can then right click the perspective extension contribution and add a view.
2. In the "Extension Element Details" (2) we can then fill the values for the view. More information about the creation of perspective extensions can be found in the "Creating custom perspectives" section (chapter "Customizing") in the document "Informatica MDM - Product 360 - <VERSION>- Knowledgebase, Installation and Customization".

Adding standard views to a custom perspective

When creating a new perspective most of the time it is necessary to add standard views like the *Article table (Items #1)* or the *Context selection* view to it. To do this, the same procedure for adding views in the perspectiveExtension has to be done.

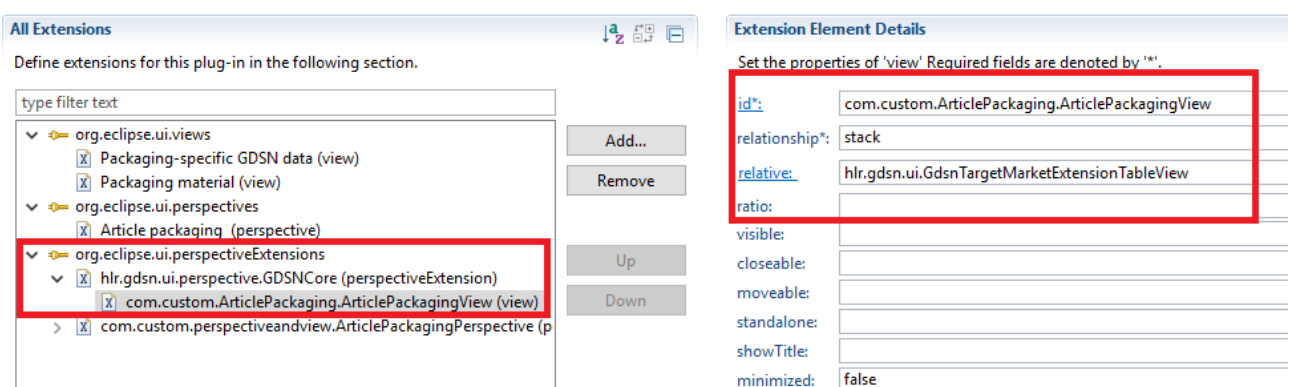
The view ids can be found in the **plugin.xml** of standard com.heiler.*.ui plugins.



Extend a standard GDSN perspective with custom views

It is possible to add a view to a standard GDSN perspective. To do this the same procedure like adding views to perspectives is used. That means only a contribution to the perspectiveExtensions extension point is needed. To add for example a view to the GDSN core perspective, the id "hlr.gdsn.ui.perspective.GDSNCore" has to be declared in the relative value in the "Extension Element Details".

All perspective and view ids from GDSN can be found in **plugin.xml** of the com.heiler.ppm.gdsn.ui plugin.



Creating additional elements in Web UI

General information

For editing detail tabs in the web, a detailed documentation can be found in the chapter "Detail Tab Definition Examples" in the document "Informatica MDM - Product 360 - <VERSION>- Knowledgebase, Installation and Customization".

To add new tabs when working with items in the web UI we have to copy the generated article.detailtab.xml from the server/configuration/HPM/webdefinitions/default directory to

the server/configuration/HPM/webdefinitions directory. That way it will not get overwritten when restarting the Product 360 Server.

We can now edit the detail tab for our needs.



When referencing Product 360 repository fields, make sure to specify the field identifier from the **custom area** but not the field type identifier from the types area.

Adding a new field

When only adding a new field to an existing module just add the tag `field` with the identifier of the new field to the already existing field group or table group.

```
[...]
<fieldGroup displaySectionWidget="true" subEntityId="ArticlePackaging">
[...]
  <field identifier="ArticlePackaging.NewFieldCreatedInTheRepository"/>
[...]
```

Adding a new module

Depending on the module, you can choose between a field group and a table group.

A field form group can be seen as a single field list. We will use the `fieldGroup` for the `ArticlePackaging` and `ArticlePackagingUOM` entity.

Make sure to use the field identifier from the custom area of the repository.

Article packaging definition

```
<definition debugId="article_packaging_tab" i18NKey="%web.article.detail.tab.packaging"
  permissionId="web.article.detail.tab.packaging" position="509" rootEntity="Article"
  ">
  <column/>
  <column>
    <enumGrouping caption="%field.targetMarket.name" enumIdentifier="Enum.Territory"
      selectable="true" value="US"/>
    <enumGrouping caption="%field.packagingType.name" enumIdentifier="Enum.PackagingTypeCode.WithOptionalCode"
      selectable="true" value="NO_CODE"/>
    <fieldGroup displaySectionWidget="true" subEntityId="ArticlePackaging">
      <field identifier="ArticlePackaging.PackagingFeatureCode"/>
      <field identifier="ArticlePackaging.PackagingFunctionCode"/>
      <field identifier="ArticlePackaging.PlatformTypeCode"/>
      <field identifier="ArticlePackaging.PackagingShapeCode"/>
      <field identifier="ArticlePackaging.DoesPackagingHaveWheels"/>
      <field identifier="ArticlePackaging.IsPackagingExemptFromRefuseObligation"/>
      <field identifier="ArticlePackaging.PackagingRefuseObligationName"/>
      <field identifier="ArticlePackaging.PlatformTermsAndConditionsCode"/>
    </fieldGroup>
  </column>
</definition>
```

```

        <logicalKey identifier="ArticleDomainType.LK.Std_LK_Text100_01"
selectable="true" value="&lt;No code&gt;"/>
        <logicalKey identifier="ArticleDomainType.LK.TargetMarket" selectable="
true" value="USA"/>
    </fieldGroup>
    <fieldGroup displaySectionWidget="true" subEntityId="ArticlePackagingUOM">
        <field identifier="ArticlePackagingUOM.UsableProductVolume"/>
        <field identifier="ArticlePackagingUOM.UsableProductVolumeUOM"/>
        <field identifier="ArticlePackagingUOM.PackagingWeight"/>
        <field identifier="ArticlePackagingUOM.PackagingWeightUOM"/>
        <logicalKey identifier="ArticleDomainType.LK.Std_LK_Text100_01"
selectable="true" value="&lt;No code&gt;"/>
        <logicalKey identifier="ArticleDomainType.LK.TargetMarket" selectable="
true" value="USA"/>
        <logicalKey identifier="ArticleDomainUOMType.LK.UOMType" selectable="
true" value="metric"/>
    </fieldGroup>
</column>
</definition>

```

The result will look like this:

Trade item lifespan	Data carrier information	Item certifications	Sustainability information	Delivery purchasing information	Packaging-specific GDSN data
Target market	USA				
Packaging type	<No code>				
Packaging feature code:	No content				
Packaging function code:	No content				
Platform type code:	No content				
Packaging shape code:	No content				
Does packaging have wheels:	No content				
Is packaging exempt from refuse obligation:	No content				
Packaging refuse obligation name:	No content				
Platform terms and conditions:	No content				
UOM types	metric				
Usable product volume (metric):	No content				
Usable product volume UOM (metric):	No content				
Packaging weight (metric):	No content				
Packaging weight UOM (metric):	No content				

For the sub entity ArticlePackagingMaterial we will use the tableGroup to show the fields in a table.

Article packaging material definition

```

<definition debugId="article_packagingMaterial_tab" i18NKey="%web.article.detail.
tab.packagingMaterial" permissionId="web.article.detail.tab.packagingMaterial"
position="510" rootEntity="Article">
  <column/>
  <column>
    <tableGroup>
      <actionPanel>
        <actionButton action="create" i18NKey="%web.article.detail.tab.cr
eate.packagingMaterial"/>
        <actionButton action="edit" i18NKey="%web.article.detail.tab.edit
.packagingMaterial"/>
        <actionButton action="delete" i18NKey="%web.article.detail.tab.de
lete.packagingMaterial"/>
        <fieldFormGroup displaySectionWidget="true" subEntityId="ArticleP
ackagingMaterialUOM">
          <field identifier="ArticlePackaging.TargetMarket"/>
          <field identifier="ArticlePackaging.PackagingType"/>
          <field identifier="ArticlePackagingMaterial.PackagingMaterial
TypeCode"/>
          <field identifier="ArticlePackagingMaterialUOM.UOMType"/>
          <field caption="%field.packagingMaterialCompositionQuantity.n
ame" identifier="ArticlePackagingMaterialUOM.PackagingMaterialCompositionQuantity"/>
          <field caption="%field.packagingMaterialCompositionQuantityUO
M.name" identifier="ArticlePackagingMaterialUOM.PackagingMaterialCompositionQuantityU
OM"/>
        </fieldFormGroup>
      </actionPanel>
      <tableDefinition i18NKey="" identifier="detail_packagingMaterial"
rootEntity="ArticlePackagingMaterialUOM">
        <field identifier="ArticlePackaging.TargetMarket" sortable="true"
/>
        <field identifier="ArticlePackaging.PackagingType" sortable="true"
"/>
        <field identifier="ArticlePackagingMaterial.PackagingMaterialType
Code" sortable="true"/>
        <field caption="%field.packagingMaterialCompositionQuantity.name"
identifier="ArticlePackagingMaterialUOM.PackagingMaterialCompositionQuantity"
sortable="true"/>
        <field caption="%field.packagingMaterialCompositionQuantityUOM.na
me" identifier="ArticlePackagingMaterialUOM.PackagingMaterialCompositionQuantityUOM"
sortable="true"/>
      </tableDefinition>
    </tableGroup>
  </column>
</definition>

```

The result will look like this:

Trade item lifespan	Data carrier information	Item certifications	Sustainability information	Delivery purchasing information	Packaging-specific GDSN data	Packaging material	Trade item hierarchy
<div><div><div><div>+</div><div>🔍</div><div>✖</div></div></div></div>							
Target market	Packaging type	Packaging material type code	Packaging material composition quantity (metric)	Packaging material composition quantity UOM (metric)			
Germany	Bag	Ceramic	123.000000	centigram			
USA	Bag	Ceramic	125.000000	centigram			

Note that the action panel, marked in red, has to be explicitly declared in the XML as `actionButton` to be shown.

Interface visibility and display rights

When creating a new view and contributing it to the extension point the display right is automatically generated and can be seen in the "Interface visibility" view in the desktop client.

For more information about the visibility of the new views and perspectives please read the "Frontend visibility" chapter in the "Informatica MDM - Product 360 - <VERSION>- Knowledgebase, Installation and Customization" document. There is a description how to manage your views and perspectives with action and visibility rights.

Limitation

It is not possible to display sub entities which exceed the 4th level repository hierarchy. When having such a case have a look at the "[Deep structures](#)"(see [page 222](#)) chapter in the GDSN Implementation Guidelines.

2.1.9.6 Data validations

Product 360 provides several ways to validate and format your product information.

- [Data model validation](#)(see [page 265](#))
- [Data quality checks by IDQ \(Informatica Data Quality\)](#)(see [page 266](#))
- [Validation and formatting during import](#)(see [page 267](#))
- [Validation and formatting during export](#)(see [page 269](#))

For GDSN those are needed to get the data correctly into the data model of Product 360, maintain the data correctly in Product 360 and export the data in the expected format of 1WS.

This chapter is refering to the "Informatica MDM - Product 360 - Desktop_ <Version>_UserManual_en.pdf" which will be called "Product 360 User Manual" in the following sections.

Data model validation

The data model validation is based on the repository data model. During importing and maintaining the data, Product 360 validates the length of character strings, the ranges of numeric values and whether the entered value is compatible to the defined enumeration if existent. In addition it checks in most cases whether it is a mandatory value or not.

See chapter "[Data model](#)(see [page 235](#))" for more information.

Data quality checks by IDQ (Informatica Data Quality)

It is possible to check your product information automatically and manual by using IDQ. Product 360 is delivered with a set of standard rules which can be adapted to your requirements by configuration. It can be that you cannot find a standard quality rule which fulfills your needs. In this case you need to create your own rule. Based on rules you can create your specific data quality configuration which is described in the "Product 360 User Manual" in chapter "Data quality checks". For GDSN the data quality checks are mainly used for data source scenario because in data recipient scenario you get only valid data from 1WS.

Creating a data quality rule

In the case you cannot define your required check by using a standard data quality rule you need to write a custom data quality rule. Please take a look at the documentation for "Data Quality" to be able to do so.

Creating a data quality configuration

You can define a specific data quality check by using a data quality configuration which is based on a data quality rule. The necessary steps to define such data quality configuration are described in the "Product 360 User Manual" in chapter "Creating quality rules".

In GDSN you often need a rule which is checking if field A is not empty, and if so field B must not be empty as well. A typical example for this is a UOM (unit of measure) value and the according UOM field like "Usable product volume" and "Usable product volume UOM". The following picture shows how the configuration is looking for this:

The screenshot shows the 'Data quality configuration' window. The left pane displays a tree structure with folders 'GDSN3', 'GDSN_Customizing', 'MyGroup (Item)', and 'LMIV'. The 'MyGroup (Item)' folder is expanded, showing a rule named 'Check items with 'Usable produ...'. The right pane shows the configuration details for this rule.

Name: Check items with 'Usable product volume' have a 'Usable product volume UOM' populated ☐ Hidden

Description: Check items with 'Usable product volume' have a 'Usable product volume UOM' populated

Rule name: Check_IfNotEmptyConditionNotEmpty

Data type: Packaging information

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Usable product volume
inCheckField	string(512)	Usable product volume UOM
inStatusMessage	string(50)	

Data type: Item

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

Data preprocessor:

Automatic data quality check execution configuration

Product 360 provides some triggers like "Export started", on which a data quality check will be automatically executed. The "Product 360 User Manual" describes in chapter "Planning quality checks" how to use the triggers.

Validation and formatting during import

In the import you can specify functions for the information to be imported. The "Product 360 User Manual" contains the list of many available import functions (import function reference) and also all functionality in the import. Furthermore you can create your own customized import function or validate your file on your own via your customized pre-import step (see documentation for "Customizing Import" and "Hot folder").

Some examples

- The value "1001.101" contained in the import file will be formatted with 2 decimal places and will show the thousand separator. The imported value would be "1,001.1".

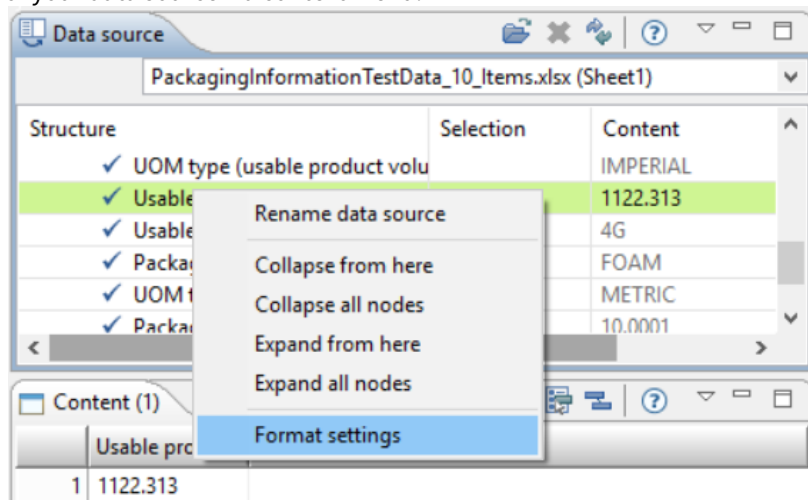
The screenshot displays the Informatica MDM Product 360 interface. On the left, the 'Data source' pane shows a file named 'PackagingInformationTestData_10_Items.xlsx'. The 'Structure' pane lists various fields with checkboxes, including 'Usable product volume'. The 'Repository' pane shows a table with columns 'Structure', 'Assignment', and 'Content'. The 'Content' column shows the value '1,001.100000' for 'Usable product volume'. The 'Field details' pane shows the mapping function 'format([Usable product volume],2,true)'.

Structure	Assignment	Content
Packaging information		
Target market	Target market	USA
Packaging type	Packaging type	Intermediate bulk container, rigid ...
Packaging feature code	Packaging feature code	Lid
Packaging function code	Packaging function code	Dispenser
Platform type code	Platform type code	ISO 0 pallet
Packaging shape code	Packaging shape code	Bar
Does packaging have wheels	Does packaging have wheels	Not applicable
Is packaging exempt from refuse obligation	is packaging exempt from refuse o...	Unspecified
Packaging refuse obligation name	Packaging refuse obligation name	obligation name 1
Platform terms and conditions	Platform terms and conditions	No exchange / no return
Packaging information UOM		
UOM type	UOM type (usable product volume)	imperial
Usable product volume	Usable product volume	1,001.100000

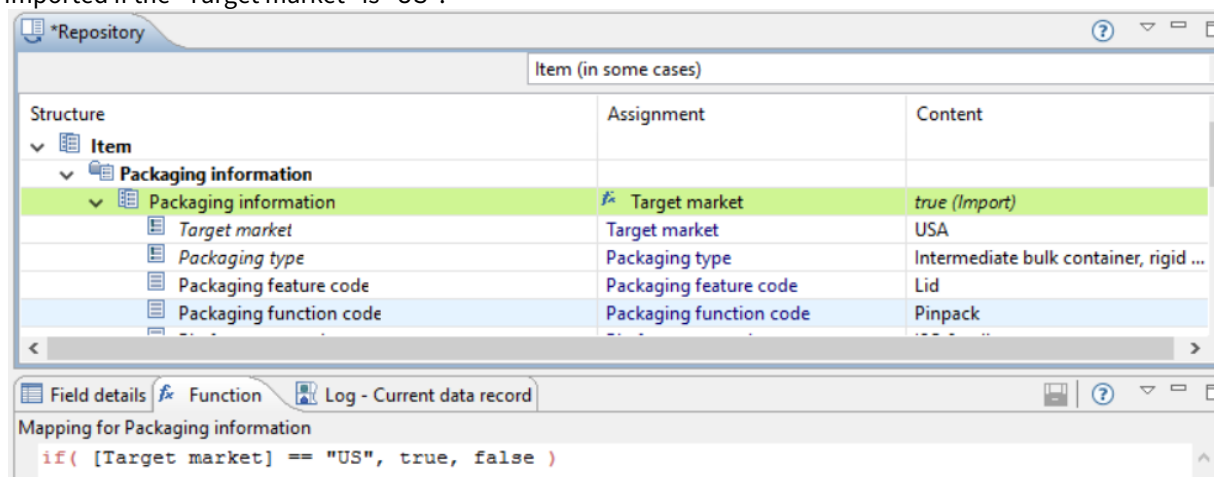
Mapping for Usable product volume

```
format([Usable product volume],2,true)
```

- Define the used pattern for decimal values and date format pattern for the whole import file or for each field of your data source via context menu.

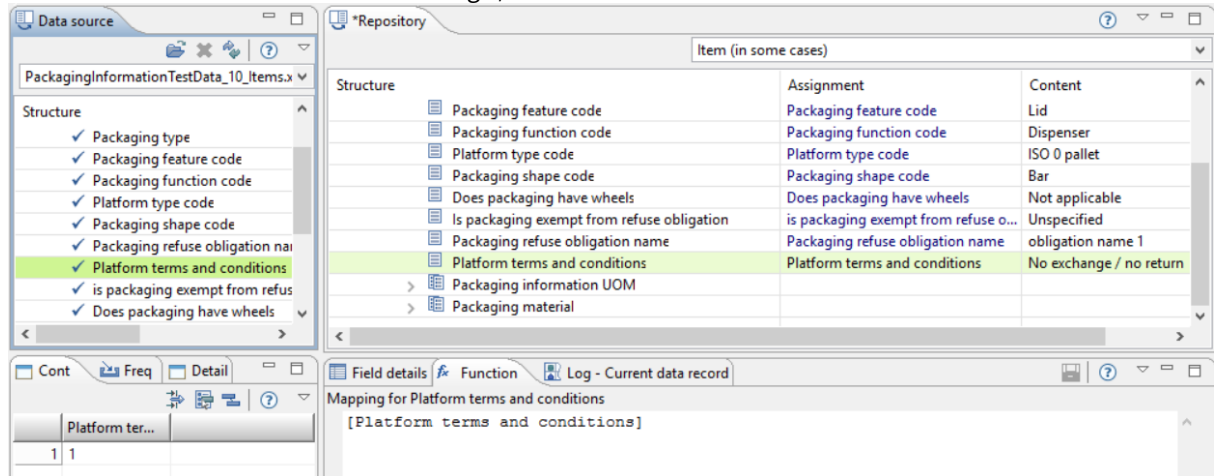


- Define if an entity should be imported by using functions. Here the packaging information will be only imported if the "Target market" is "US".



- It is recommended for fields with enumeration to import their values by using the defined key in the repository. In the example the "Platform terms and conditions" value in the Data Source is "1" but the

shown value in Product 360 is "No exchange / no return".



Validation and formatting during export

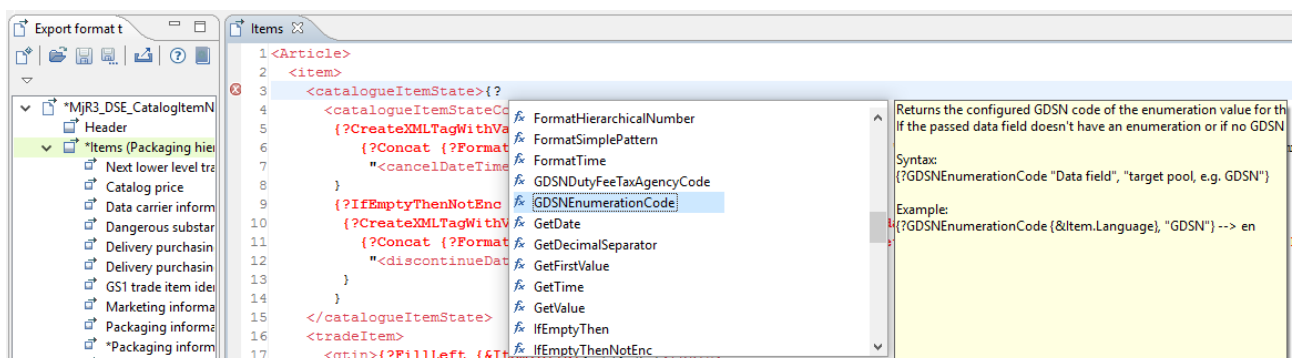
The export can check the data for specific criteria and also format the data in the expected format. Those configurations have to be done within the export format template.

Validation rules for data fields

You can define validation rules for each data field. Those have the focus on the according field value and cannot take references to other field values. In the chapter "Validation rules for data fields" of "Product 360 User Manual" it is described how to use this kind of validation.

Validate and format by export functions

The export comes with many functions where the Product 360 User Manual describes many of them in chapter "Export function reference". The "Product 360 User Manual" describes many of them in chapter "Export function reference". The currently available list of functions, including the GDSN specific functions, can be displayed in Export template perspective of Product 360 directly. You can access those via typing "?" and pressing "Ctrl+Space".



By using the correct function you can ensure that the information is exported in the expected format. Please format all your numbers and enumeration values as shown below. In addition always format your date and

boolean values as well. You can use the delivered export templates as a reference in case you are unsure how to export it correctly.

```
<packagingMaterial>
  {?CreateXMLTagWithValue {?EnumerationKey (&Packaging material.Packaging material type code)}}
  , "<packagingMaterialTypeCode>?</packagingMaterialTypeCode>"
  {?CreateXMLTagWithValue {?FormatDecimal (&Packaging material.Packaging material composition quantity (UOM type))}, ".", 6),
  "<packagingMaterialCompositionQuantity measurementUnitCode=\"(&Packaging material.Packaging material composition quantity UOM (UOM type).Code (Unit system))\">
  ?
  </packagingMaterialCompositionQuantity>"}
</packagingMaterial>
```

You can find more details in chapter "[Technical details\(see page 84\)](#)" of GDSN Accelerator documentation.

Validate the exported file

The GDSN communication is based on XML files. Therefore each message has defined XSD schemata which the export can use to validate the exported file. This validation will fail in case of exporting data in a not expected way. If an XSD validation error occurs, the export will be canceled and you can check if you might missed to format one of your values. Please ensure that you are not exporting an XML tag without value because this is leading to an XSD failure as well.

Limitations:

- If the export fails due to an XSD validation error, the file will not be saved. This could be a problem when the XSD validation error message is not clear. This is especially the case for flex attributes which have generic XML tags and so the error messages are not unambiguous. A possible workaround is to setup a post export step (at the very beginning) which copies the file to a local directory. Then you can check the line number from the error message and see what's really wrong.
- If no items are exported because the assortment is empty or all of the items are "sorted out" due to a failing DQ rule per item, the XML file will not contain any item. This leads to an XSD validation error and the file will not be sent to the GDSN pool. Although the behavior is correct but it is not obvious to the user what happened.
- If the export fails due to an XSD validation error the process overview shows "Cancelled by user". This is currently a technical restriction which has no bad influence on any data but can be confusing.

You can find more details in chapter "[Technical details\(see page 84\)](#)" of GDSN Accelerator documentation.

2.1.9.7 Communication

The provided approach for the communication choreography to the according 1WS data pool of the GDSN Accelerator is file based. Although in case of using another certified data pool for GDSN this could be differently. That's why the following chapters are covering adjustments concerning the choreography supported by the GDSN Accelerator as well as hints in case of using a different certified data pool. Furthermore we assume that you already have a GLN (Global Location Number) as well as all needed connection data to the data pool.

Communication to the 1WS data pools

In case you want to send more or less GDSN related information to the data pool in the data source scenario or if you want to receive more or less GDSN related information from the data pool in the data recipient scenario you need to adjust the used mechanism for the communication accordingly.

In this chapter it will be explained, how those adjustments should be done and what needs to be considered.

Resources

There are several documents on which these sections are referring to. Please make sure that you are familiar with those documentations.

- "Informatica MDM - Product 360 - Desktop_ <Version>_UserManual_en.pdf" which will be called "Product 360 User Manual" in the following sections.

Adjust or generate an import mapping (Data Recipient)

In data recipient scenario product information will be sent by using a CIN file from the data pool to Product 360 via different components of the GDSN Accelerator. If using the IM connection, the received CIN file will be forwarded to the hotfolder of Product 360. But if you use the DSE connection, hierarchical product information will be flattened before the file is transferred to the hotfolder of Product 360. Because of this you need to take the accordingly adjusted XML to create an import mapping.

For being able to import the CIN file correctly, you should have a working 1WS pool connection where you can receive an example CIN file. In case of not having a working connection you can simulate the process through B2B by using your own generated XML file. The "Product 360 User Manual" describes in chapter "Data import" the needed *Import* perspective of Product 360 and the available functionalities to define the import mapping. With this knowledge you will be able to define the needed import mapping.

Please consider the according chapters "[Data validation\(see page 265\)](#)" and "[Testing\(see page 279\)](#)" of this documentation.

Example

For a better understanding we will assume that we will have to add the mapping of a new GDSN Module. For this example we will take the "MicrobiologicInformationModule" of the IM data pool.

1. Pick up a CIN file in the hotfolder of Product 360 which contains product information for the "MicrobiologicInformationModule".

The product information could look like this:

```
<attrGroupMany name="psychotropicSubstance">
<attrGroupMany name="foodAndBevMicrobiological">
  <row>
    <attr name="organismCode">BACILLUS_CEREUS</attr>
    <attrQual name="organismMaximumValue" qual="80">973235.8</attrQual>
    <attrQual name="organismReferenceValue" qual="80">939344.1</attrQual>
    <attrQual name="organismWarningValue" qual="80">585250.1</attrQual>
  </row>
  <row>
    <attr name="organismCode">CAMPYLOBACTOR</attr>
    <attrQual name="organismMaximumValue" qual="2Q">80498.8456</attrQual>
    <attrQual name="organismReferenceValue" qual="2Q">68340.1982</attrQual>
    <attrQual name="organismWarningValue" qual="2Q">66703.1465</attrQual>
  </row>
</attrGroupMany>
</attrGroupMany name="physioChemicalProperties">
```

2. Create a new import project with the XML file of step 1 in the *Import* perspective of Product 360.

3. Navigate in the *Data Source* view to the "foodAndBevMicrobiological" flex attributes. The XML path has been looked up in the chapter "Analyze requirements(see page 222)".

Structure	Selection	Content
flex		
attr (1)	1/193	true
attrGroup (1)	1/2	
attrGroupMany (2)	33/85	
name (9)		foodAndBevMicrobiological
row (2)	2/2	
attr (4)		CAMPYLOBACTER
name (10)		organismCode
attrGroupMany (3)		
attrMany (6)		
attrQual (6)	1/3	80498.8456
name (42)		organismMaximumValue
qual (12)		2Q
attrQualMany (7)		
attrGroup (5)		
attrQual (10)	1/24	7189795167338534963346.1
attrQualMany (13)	1/45	
attrMany (13)	1/47	
cancelDate		

4. Map the fields accordingly to the Product 360 data model.

Structure	Selection	Content
flex		
attr (1)	1/193	true
name (2)		IsTradeternNonPhysical
attrGroup (1)	1/2	
attrGroupMany (2)	33/85	
name (9)		foodAndBevMicrobiological
row (2)	1/2	
attr (4)		BACILLUS_CEREUS
name (10)		organismCode
attrGroupMany (3)		
attrMany (6)		
attrQual (6)	3/3	585250.1
name (42)		organismWarningValue
qual (12)		80
attrQualMany (7)		
attrGroup (5)		
attrQual (10)	1/24	7189795167338534963346.1
attrQualMany (13)	1/45	
attrMany (13)	1/47	

Structure	Assignment	Content
Item		
Microbiological information		
Target market	name (9)	true (Import)
Organism code	targetMarket (2)	USA
UOM	attr (4)	Bacillus cereus
UOM type	"imperial"	imperial
Organism maximum value	name (42), attrQual (6)	973,235.800000
Organism maximum value UOM	name (42), qual (12)	pounds per square inch absolute
Organism reference value	name (42), attrQual (6)	939,344.10
Organism reference value UOM	name (42), qual (12)	pounds per square inch absolute
Organism warning value	name (42), attrQual (6)	585,250.10
Organism warning value UOM	name (42), qual (12)	pounds per square inch absolute

5. Execute the import and check if the product information is available in Product 360 as expected.
6. Execute the complete workflow for receiving a CIN of the GDSN Accelerator.

Adjust the export template (Data Source)

In this chapter we will explain, how to adjust the export format template to send product information to the 1WS pool with the GDSN Accelerator. Therefore we need a XML example to be able to see in which XML structure that product information has to be sent. Therefore you can use the XSD files or download an example file from 1WS. Please note that the XML path for the DSE connection will have an additional "Article" tag, because we are exporting a flattened hierarchy, which will be hierarchically transformed before sending to the data pool.



You can generate a XML file by using the provided (or downloaded) XSD files to be able to create an import mapping by various third party tools. But usually you can only take the generated structure and need to modify the attributes regarding valid values of GDSN.

You can find general information about how to work with the Product 360 Export in the "Product 360 User Manual" chapter "Export". Furthermore also consider the according chapters regarding export in "[Data validation](#)(see page 265)" and "[Testing](#)(see page 279)" of this documentation.

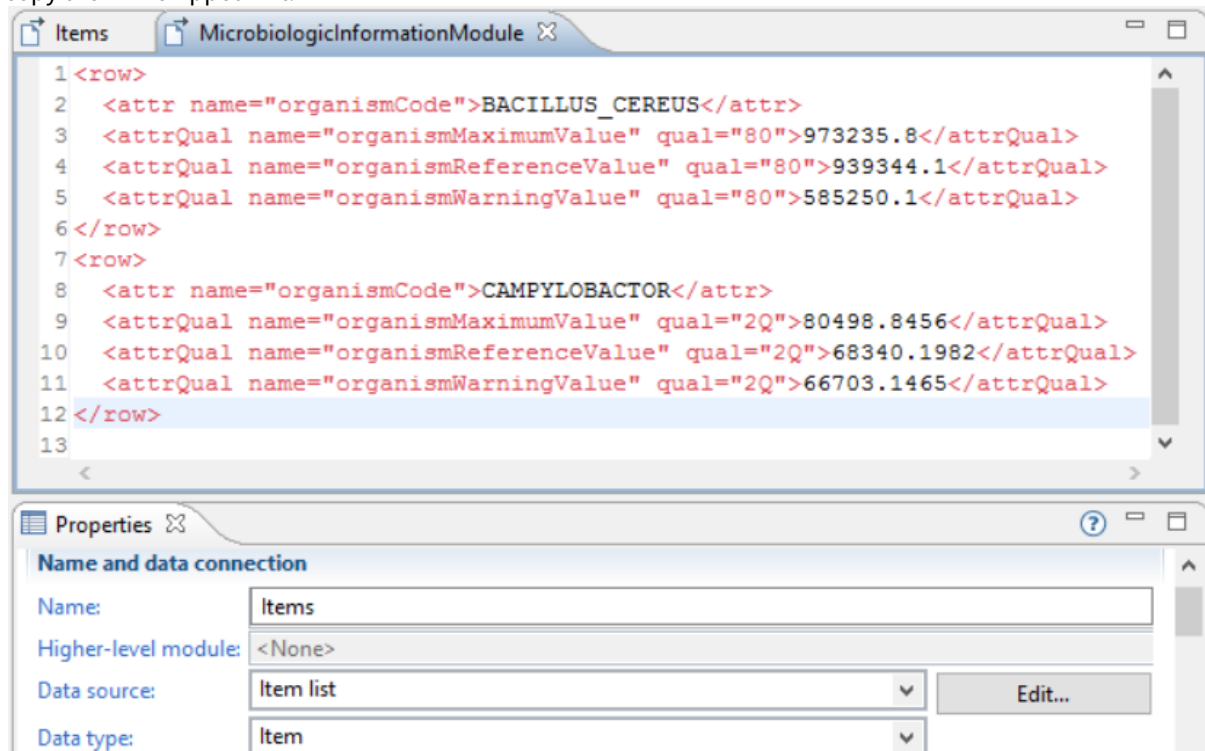
Example

As example we assume that we've added the GDSN module "MicrobiologicInformationModule" to the Product 360 data model and want to send those product information to the 1WS pool by using the IM connection.

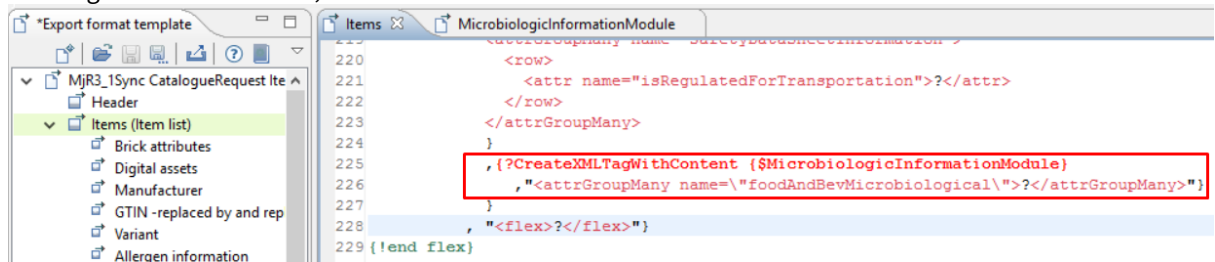
1. Get a XML snippet of the "MicrobiologicInformationModule" in the "CatalogueRequest" message and ensure that it contains valid data. Usually you find GDSN modules mapped to "flex" attributes when using IM.

```
<attrGroupMany name="psychotropicSubstance">
<attrGroupMany name="foodAndBevMicrobiological">
  <row>
    <attr name="organismCode">BACILLUS_CEREUS</attr>
    <attrQual name="organismMaximumValue" qual="80">973235.8</attrQual>
    <attrQual name="organismReferenceValue" qual="80">939344.1</attrQual>
    <attrQual name="organismWarningValue" qual="80">585250.1</attrQual>
  </row>
  <row>
    <attr name="organismCode">CAMPYLOBACTOR</attr>
    <attrQual name="organismMaximumValue" qual="2Q">80498.8456</attrQual>
    <attrQual name="organismReferenceValue" qual="2Q">68340.1982</attrQual>
    <attrQual name="organismWarningValue" qual="2Q">66703.1465</attrQual>
  </row>
</attrGroupMany>
<attrGroupMany name="physioChemicalProperties">
```

2. Create a sub-module for the GDSN "MicrobiologicInformationModule" in the export format template and copy the XML snippet in it.



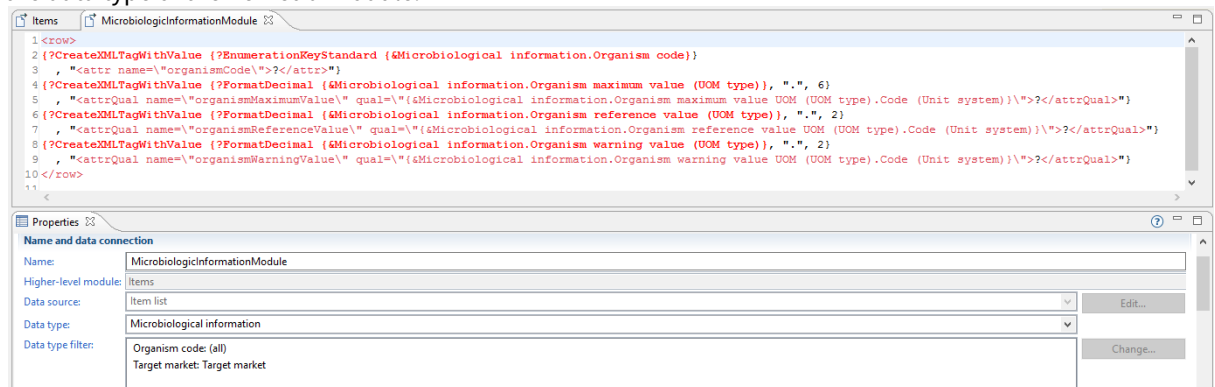
- Call the sub-module in the "Items" module of the export format template. Please note that there is no ordering of "flex"-Attributes, so we can add it in the end.



- Execute an export against the data pool with a single item and check if the product information for the GDSN module really was in the generated file and if the response from the data pool was without errors.

	Informa...	Trade partner	Target...	Message type	Context	Response type	Executed operation
1	Inform...	MJR3 Pool	USA	Item response	<no context>	Acknowledgement	Modify
2	Inform...	MJR3 Pool	USA	Item transferred	<no context>		Modify
3	Inform...	MJR3 Pool	USA	Item registry response	<no context>	Acknowledgement	Add

- Exchange the static values with the field values and test with a single item against the data pool. Now adjust the data type of the new sub-module.



- Test the template against the data pool with several items as explained in chapter "Testing(see page 279)".



Please keep in mind that the maximum depth of the hierarchy taken by the hierarchy data sources in the export is limited by 10 levels. In case of using more levels, the standard assumes that you have an endless loop.

Adjustments to B2B Data Exchange GDSN Accelerator

GDSN integration requirements can vary from customer to customer. These requirements should be considered carefully in order to understand if and to what extent B2B Data Exchange GDSN Accelerator can support.

On the one hand, a new requirement or a change to the GDSN data model might require a simple adjustment to one or more of the B2B Data Exchange GDSN Accelerator components. Moreover, sometimes no

adjustments will be needed at all. On the other hand, there are cases in which a significant effort is required in order to adjust the B2B Data Exchange GDSN Accelerator to meet the new requirement or change.

This section describes some of the common use case and their implication on the B2B Data Exchange GDSN Accelerator:

Q: How do XSD changes in modules or fields affect B2B Data Exchange GDSN Accelerator?

A: Let's divide the question into different use cases:

- If a new module is added to the GDSN data model → an updated XSD with the new module should be deployed to the appropriate B2B DT Services.
- If a field in the XSD got changed → an updated XSD with the new module should be deployed to the appropriate B2B DT Services.
- If a new field of an existing module is added in Product 360 without XSD change → no change required in B2B.
- If an existing field is removed in Product 360 without XSD change → no change required in B2B.



In general, in case of a requirement leading to a change in any of the XSDs that are used in B2B Data Exchange GDSN Accelerator the updated XSDs should be deployed to the appropriate B2B DT services.

Q: In case there is a requirement to specify or parameterize the file name of incoming files from the data pool, how can it be achieved?

A: It depends what kind of change or parameterization is required. In general, the file name can be specified as part of the DX endpoint properties. In case the change is to add a fixed value or a parameter that is already used in the current implementation, then this can be achieved by adding it in the file pattern within the DX endpoint option.

For example, the image below shows how the "eventId" is added to the file name of incoming files from the data pool.

Edit Endpoint endpoint_gdsn_message_from_dx_to_PIM

General **File Send Options**

☒ Use endpoint root directory

Outgoing files path

* File pattern

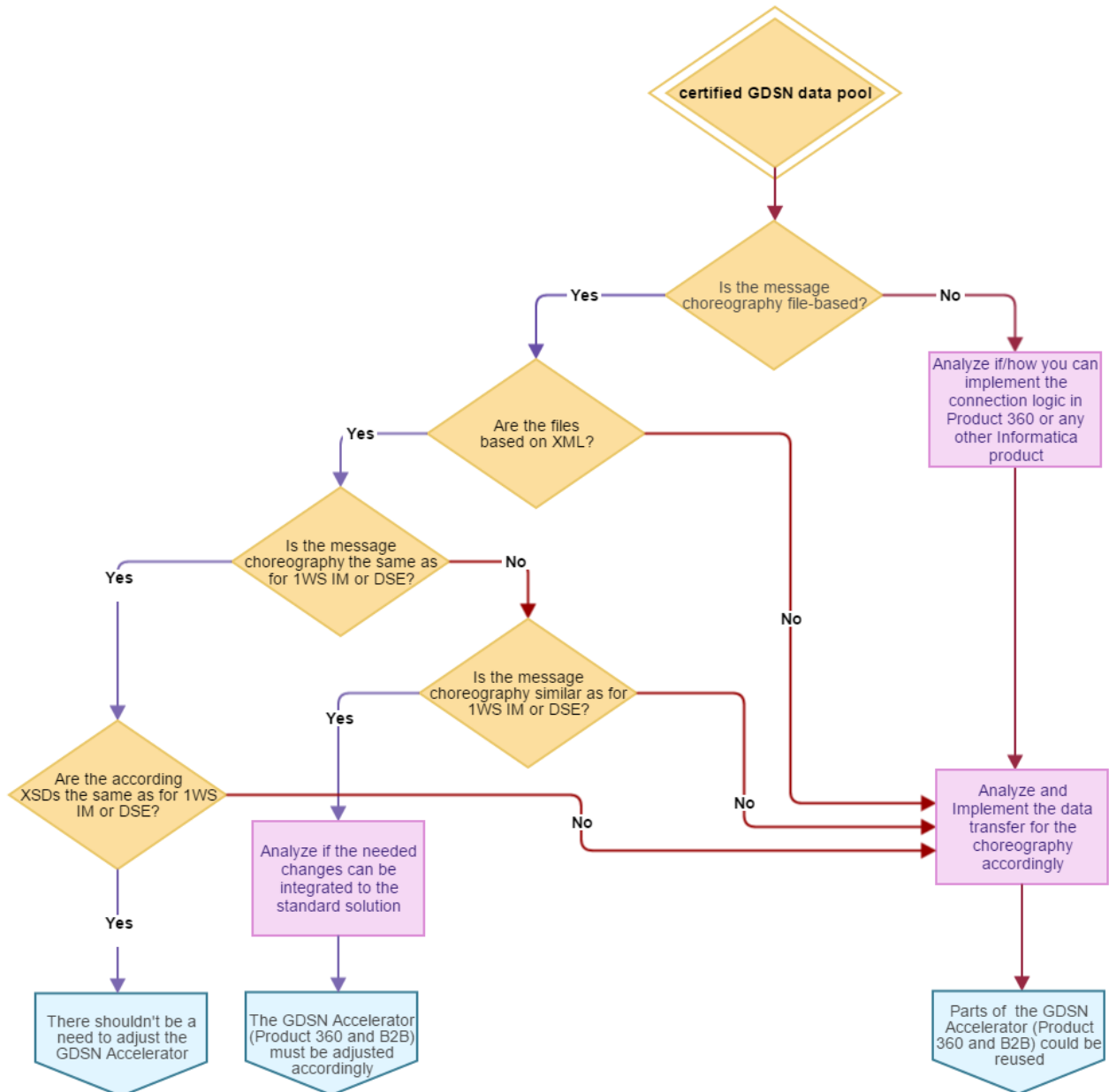
In case the parameter is not used in the current implementation, then this change will require an analysis and development effort to understand where the required parameter can be captured from and then implement it in the PowerCenter mapping and/or DT service.

Communication to other data pools

The GDSN Accelerator relates to the 1WS data pool. 1WS is one of round about 30 GDSN certified data pool providers. In case of not using the 1WS data pool, you might be able to reuse parts or the whole GDSN Accelerator in order to connect to your data pool. This chapter is more a mind map about questions you have to ask yourself in order to implement the communication to a different pool. You should be familiar with your

data pool and how the message choreography is working to be able to evaluate the similarities and differences to the 1WS approach and the provided GDSN Accelerator.

The following diagram gives you a high-level overview of what to review regarding the provided GDSN Accelerator and how many changes you can expect.



File-based communication choreography

If you need to use a file-based communication choreography then you have high chances that you can manage your needed behavior by adjusting the GDSN Accelerator on Product 360 and B2B side. Please note that the implementation can still turn out in a high effort.

The standard assumes different conditions for the communication which is differing for DSE and IM solution. You need to ensure that your data pool will support the same conditions.

IM (Data Source)

- The standard assumes, that all your items are in the master catalog. Although service API call for the publication status would work for all messages except the "ItemAuthorizationResponse".

DSE (Data Source)

- The standard assumes, that all your items are in the master catalog. If this is not the case, the service API call for the publication status will fail.
- The standard assumes, that the GTIN is unique in the master catalog. If this is not the case, the first found item with this GTIN will be taken.
- The standard assumes, that the item hierarchy will be exported flattened by Product 360.

DSE (Data Recipient)

- The standard assumes, that the GDSN information is not arriving hierarchical in Product 360. They must be flattened in order to be able to import them.

B2B Data Exchange (Communication protocols)

Informatica MFT product is responsible for the connection with the data pool. The most common communication protocol is AS2, yet MFT can support many types of protocols. You need to understand with the relevant data pool which file transfer protocol should be used to send the file to it and then configure it accordingly in MFT. The communication protocol has no impact on B2B Data Exchange and PowerCenter implementation, only on the MFT.

General checks

- Are the used files for the communication based on XML? If not then you have to create the according import mapping or export format template on your own.
- Is the communication based on the same or similar XSD files as for 1WS IM or DSE? If similar, then you can adjust the according import mapping or export format template of the GDSN Accelerator.
- Are the valid value lists and the unit system the same or similar as those for 1WS IM or DSE? If similar, then you can extend the enumerations in the repository.
- Are the needed data fields qualified as they should be?
- Do you need further data quality checks?
- Does the added export and import functions for the GDSN Accelerator fit your needs as well?

Different communication choreography

First, you need to analyze how the communication choreography can be implemented. You might need to implement a new view in Product 360 to handle your specific data connection or you can consider using another Informatica product to do the communication accordingly. Next, you need to take care of the data transfer which needs to be done in the defined communication choreography. It is very likely that the communication choreography completely differs from the used choreography of the GDSN Accelerator.

- Check if the standard enumerations in the repository match the requirements of your GDSN certified data pool.
- Check if the standard repository fields match the requirements of your GDSN certified data pool.
- Check which target markets you need.
- Check if the standard data quality checks of the GDSN Accelerator still fit your needs.
- Check which Product 360 components can support your requirements.
- Check if an Informatica product can help you with your requirements.

Data Source

- How can a user see the current status of an item or hierarchy in your GDSN certified data pool?

- Can you modify or correct the data which is already in your GDSN certified data pool?
- How can you create an automatic message process?

Data Recipient

- How can you import the data of your GDSN certified data pool?
- How can you provide feedback on the received data?

Example Electronic Commerce Council of Canada Network Services (ECCnet) - Recipient

In this example an analysis of the ECCnet data pool for the data recipient scenario will be shown which describes changes you need to consider.

Connection logic and data transfer

In contrast to the GDSN Accelerator that is using a file transfer via B2B, the communication of ECCnet is based on SOAP calls. In this case the mechanisms of the export to send data to the pool and the import to receive data from the pool cannot be used and the following two questions have to be answered:

- Source scenario: How to collect the data and send it to the pool?
- Recipient scenario: How to receive data from the pool and process the data?

In the recipient scenario the differences in the message choreography helped as described in the next section.

The "import" of the data in case of ECCnet has been done by item processors reading the data from the SOAP call and writing them to the `detailModels` using `put` commands.

Another point that came up while implementing the ECCnet Accelerator were data transformations. For example think about units: the data pool sends a string based code but in the data model (`detailModel`) a `UnitProxy` is stored. The GDSN Accelerator makes use of the capabilities of the import and export components, which provide some built-in transformations which can be extended by using import and export functions. You need to find a place for these transformations in your specific data processing.

Since we want to implement the ECCnet data recipient scenario, we assume to get valid data and don't need to care about (DQ based) validations. However, in case you need validations (in the source scenario for example) you might be able to reuse existing DQ rule configurations. Anyway, in general validations have to be analyzed and needed validations have to be created. @

Message choreography

The ECCnet choreography differs from the assumed choreography of the GDSN Accelerator. ECCnet uses a pull mechanism, while the GDSN Accelerator assumes a push mechanism.

In ECCnet first of all you have the possibility to search for products you want to get in your system. So a search UI is needed as well as an implementation of the according calls to the pool. From the search result you can "subscribe" items. As a result you will get the data, and the data pool keeps in mind that you have this data. However, this does not mean that you will get updates after the item changed in the data pool automatically. Remember, ECCnet uses a pull mechanism. When you want to update the data in your system you have to actively trigger a call to the data pool. For this action a UI is needed and maybe some tasks or workflows might come in handy.

In GDSN the requirement is to keep track of the messages that were sent and received. Additionally you want to know if something went wrong while the message has been processed by B2B. In ECCnet there is much less information. Here an item can be marked as created, updated and might be marked as deleted in the context of ECCnet. That's the reason why ECCnet does not reuse the `PublicationStatus` but instead creates additional `ArticleLog` entries.

Reusable parts

Regarding the communication nothing from the GDSN Accelerator can be reused.

ECCnet provides a mapping from ECCnet attributes to GDSN attributes. Therefore the major part of the data model and the maintenance UI can be reused. However, this doesn't mean that an analysis of the data model is unnecessary. There are many small differences like field lengths, sometimes data types and so on to consider and to adjust the data model accordingly. For example ECCnet only supports the Canadian target market, so all target market logical keys get the default value "Canada" and are deactivated. Also different codes for the units make a new unit system for ECCnet necessary.

2.1.9.8 Testing

There are several ways to ensure your changes are correct. In this chapter we will describe how your test data has to look like to test the correct behavior in Product 360. Those can be used to test different modules of Product 360, like import, maintain functionality, data quality checks and exporting the data.

Test data

The test data should cover various data constellations. Therefore you always have to create the data with following specifications:

- Always test with several datasets of each entity of your created/adjusted data model
- Always test limits of your created/adjusted data model
- Always test with data which are valid and with data which are invalid (also known as good and bad test data). Check if the invalid test data lead to the expected severity (error/warning/info).
- Always create some "complete" items, that means items containing all data you want to send to the GDSN pool.

You usually can reuse the test data for testing all modules of Product 360 except invalid test data.



If you enhance your test data with a remarks field you can note your expected result in it.

Test data model

In the chapter "[Data model \(see page 235\)](#)" we introduced how to add or adjust GDSN modules and their fields in Product 360 which includes a "[Data model check list \(see page 0\)](#)". This check list ensures a correct defined data model. It's also checking if the changes in the Product 360 data model are visible by the generic functionality of Product 360. For the mentioned check list you should create test data for exhaustive testing.

For example you can create an Excel file as import file, like shown below for the fields Packaging type, Packaging feature code, Usable product volume and Usable product volume UOM of the sub entity ArticlePackaging, which is representing the GDSN Module PackagingInformationModule:

Item no.	Target market (LK) (Enumeration)	Packaging type (LK) (Enumeration)	Packaging refuse obligation name (String)	UOM type (LK) (Enumeration)	Usable product volume (Decimal value)	Usable product volume UOM (Enumeration)	Remarks
TestPackInfoMod_1	US	AA	obligation name	IMPERIAL	1001.101	4G	Should pass
TestPackInfoMod_1	US	AA	obligation name	IMPERIAL	123456.12345		Should pass, but data quality check will fail because of missing UOM
TestPackInfoMod_1	US	CNG	obligation name	IMPERIAL	9999999999.999999	LTR	Should pass
TestPackInfoMod_1	US	CNG		METRIC	1253.6264	LTR	Should pass
TestPackInfoMod_1	ES	PUG	<String of 200 characters>				Should pass
TestPackInfoMod_2	US	AA		IMPERIAL	0		Should pass, but data quality check will fail because of missing UOM
TestPackInfoMod_3	BB	ABC	obligation name				Should fail due to not existent Packaging type
TestPackInfoMod_3	BB	STR	obligation name	METRIC	1000000000.000000	LTR	Should fail due to a range failure at Usable product volume
TestPackInfoMod_3	BB	STR	<String of 201 characters>				Should fail due to a range failure at Packaging obligation name

Item no.	Target market (LK) (Enumeration)	Packaging type (LK) (Enumeration)	Packaging refuse obligation name (String)	UOM type (LK) (Enumeration)	Usable product volume (Decimal value)	Usable product volume UOM (Enumeration)	Remarks
TestPackInfoMod_3	BB	STR	obligation name	TEST	234.3	4G	Should fail due to not existent UOM type

(Note: this table does not contain all scenarios to test)



If you write the max value "9,999,999,999.999999" for decimal values, Product 360 is actually transforming it to "9,999,999,999.999998".

Test import

Create an import mapping for your excel file and execute the import. Ensure that all expected errors are printed in the import log. Please check the created data after for correctness.

Test export

Create an export to check if you can export the data as expected. This also ensures that all fields are visible with the needed qualification.



This test is usually not made with the GDSN XML schemata.

Test Service API

If you are planning to use the Service API for your fields, ensure that you have tested it.

Test data quality

This chapter is only needed in case you created data quality rules or data quality configurations.

Testing created data quality rules

There are many ways to test your created data quality rules. For example you can test the created data quality rules within the Informatica Developer. The necessary steps are explained by using the "IfNotEmptyConditionNotEmpty" rule:

1. Create an Excel file which contains all input ports and an expected behavior as description.

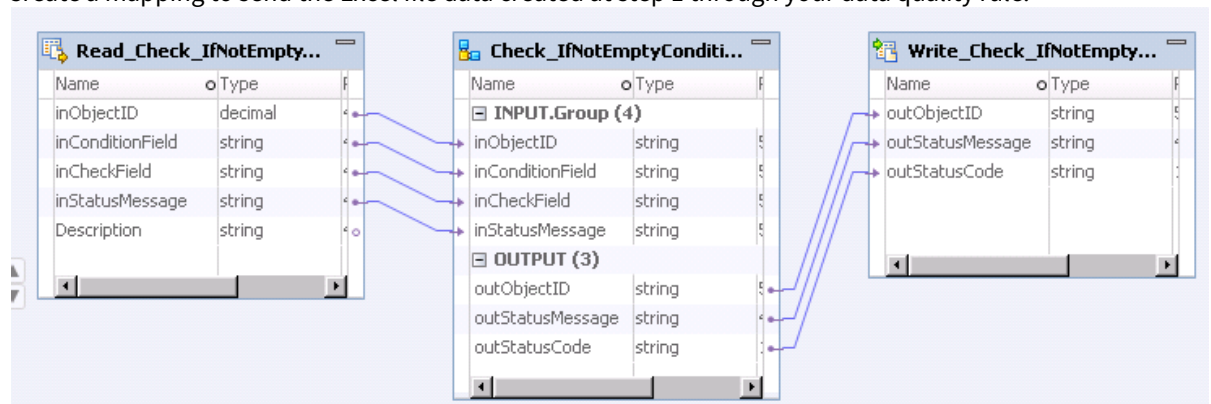
inObjectID	inConditionField	inCheckField	inStatusMessage	Description
10001			GDSN1	both inputs empty, ok
10002		somevalue	GDSN1	first input empty, ok
10003	somevalue		GDSN1	first input not empty, but second one, error
10004	somevalue	another	GDSN1	both inputs not empty, ok
10005	somevalue			first input not empty, but second one, with default error message
10101	somevalue		GDSN1	aggregated, not ok
10101	somevalue	another	GDSN1	

(Note: not all possible test scenarios are covered here)

2. Create an Excel file which contains the expected output after the data quality rule was running.

outObjectID	outStatusMessage	outStatusCode
10001	No Error	1
10002	No Error	1
10003	No Error	1
10004	GTIN is required but not provided	0
10005	inCheckField must be empty because inConditionField is not empty	0
10101	GTIN is required but not provided	0

3. Create a mapping to send the Excel file data created at step 1 through your data quality rule.



4. Compare the expected result from step 2 with the actual result of step 3.

Testing created data quality configuration

Depending on your data quality check, you can reuse the data you already created for the data model test. But you might need to extend those with further data constellations.

A recommended way to test if your data quality configuration is working as expected is:

1. Think about all possible data constellations and create your data accordingly.
2. Execute the data quality check for your configuration on the data of step 1.
3. Evaluate if your expectation and the actual result is equal.

In case the data quality configuration result is differently as your assumption, please check following:

1. Are the correct fields mapped to your data quality configuration?
2. If existent, is the correct qualification of the fields used?
3. If existent, is the port "InObjectID" mapped?
4. Did you use the correct data type?
5. In case you are using a sub entity of item as data type, then there must be one data entry for this sub entity to execute the rule.
If you want to ensure that a value is in a specific field of a sub entity use the item as data type instead of the sub entity and qualify the field accordingly. You could also implement a data quality rule which is checking the field via service API.
6. Is the used data quality rule the correct rule for your validation? Please check the description of the rule again.

Test against certified GDSN pool (Data Source)

If you have done all the testing and you ensured that all data is correct in Product 360 you need to enhance your export format template for GDSN accordingly. In chapter "[Data validation\(see page 265\)](#)" is already described how to manage the correct formatting of each field in the export.

You can reuse your test data created for the data model testing but in addition, you have to ensure that all mandatory fields of GDSN are maintained. Otherwise the XSD validation will fail during export.

When sending data to the certified pool, ensure that you have the data to export validated. The exported data should get accepted by your data pool. In case of using 1WS data pool, you can verify the correctness of the data in the XML file by opening the "Publication status" view in Product 360. You will see an entry with "Item response" as message type and the according response type. If the response type is "Acknowledgement", then your changes seem to work. Please check the exported file to ensure that data for changes are exported.



	Message type	Information provider	Created on	Trade partner	Target market	Context	Severity	Response type	Executed operation	Event ID
1	Item registry response	OurselfIP	8/21/2015 4:54 AM	PoolGLN	USA	<no context>		Acknowledgement	Add	81,072
2	Link response	OurselfIP	8/11/2015 12:50...	PoolGLN	Worldwide	11900000012514		Acknowledgement	Add	82,006
3	Link transferred	OurselfIP	8/11/2015 12:49...	PoolGLN	Worldwide	11900000012514			Add	82,005
4	Item response	OurselfIP	8/10/2015 6:36 AM	PoolGLN	USA	<no context>		Acknowledgement	Modify	81,075
5	Item transferred	OurselfIP	8/10/2015 6:35 AM	PoolGLN	USA	<no context>			Modify	81,074

In case of an exception as response type something is wrong at your data. Please evaluate what the exact error message is saying. Sometimes it is already helpful to just investigate the exported XML file for "strange" things.

Publication status (5)							Publication messages (1)	
11900000012606							OurselfIP - PoolGLN - USA - Item response - <no context>	
Message type	Target market	Context	Response type	Executed operation	Event ID		Message	Code
1 Item transferred	USA	<no context>		Modify	85,016			
2 Item response	USA	<no context>	Exception	Correction	85,018			
3 Link transferred	Worldwide	11900000012590		Add	85,035			
4 Link response	Worldwide	11900000012590	Acknowledgement	Add	85,036			
5 Item registry response	USA	<no context>	Acknowledgement	Add	82,046			

Please always publish your data to the 1WS data pool. This is needed since some validations are only running during the publish process.

2.1.9.9 FAQ

FAQ

Q: How do we add missing units of measurement to the GDSN unit system?

A: Open a ticket at the Informatica support. Informatica will provide a database script which adds the needed units.

Q: Why does the deletion of a subentity of "Article" doesn't work via Service API when using a reserve logical key as qualification filter?

A: When deleting a subentity, the used `qualificationFilter` arguments must not only unique in the entity path, it must be also unique of all subentities of the target entity. For example you want to delete all your `ArticleIngredient` objects which have a specific qualification, the statement could look like this:

```
http://localhost:1513/rest/V1.0/list/Article/ArticleIngredient/byCatalog?
catalog=Catalog_Test_RestAPI&qualificationFilter=res_LK_Text100_01("DEFAULT")
```

The problem is that several logical keys are using "res_LK_Text100_01" as Object Name, to be exact the logical keys of entity `ArticleIngredient` and `ArticleIngredientComponent`. To prevent this define an Alias for your logical key. In case this is not solving your problem, please contact the Informatica support.

2.1.10 GDSN Customization

2.1.10.1 Customize how items are found for the set publication status rest call

As you've read in the chapter GDSN Accelerator message choreography the communication between PIM and the GDSN data pool is based on XML messages. The history of these messages are stored at the item in form of a list of publication status entries.

A publication status is written by a Rest call executed by the Power Center workflows in B2B.

In order to write a publication status, first the item has to be found to which the status should be added. The standard implementation assumes

- that only items in the Master catalog are used with GDSN

- that the GTIN of the item is unique in the system

If these assumptions are not true, it is possible to write an extension for the extension point "com.heiler.ppm.gdsn.server.publicationStatusItemFinder" in order to add a custom logic to find the correct item.

Extension

A possible extension can look like this

```

1  <plugin>
2  ...
3    <extension
4      id="com.heiler.ppm.publicationStatusFinderMasterAndSupplier"
5      point="com.heiler.ppm.gdsn.server.publicationStatusItemFinder">
6      <PublicationStatusItemFinder
7        class="com.heiler.ppm.custom.gdsn.server.CustomPublicationStat
8        usItemFinder"
9        order="100">
10     </PublicationStatusItemFinder>
11   </extension>
12   ...
13 </plugin>

```

Please also read the extension point description, available in the plugin.xml editor in eclipse.

The contribution consists of an implementing class and the order. If multiple extensions are found, the one with the highest order will be used. If a non numeric value is given, it will be considered a low value.

Implementation

The class with the custom logic has to implement the interface PublicationStatusItemFinder.

```

public interface PublicationStatusItemFinder
{
    EntityProxy findItemByEntityItemReference( EntityItemReference entityItemReference,
String targetMarket,
                                           ChannelProxy channel, PartyProxy
informationProvider,
                                           PartyProxy tradePartner, String
messageType, String publicationMarket,
                                           String context )
        throws CoreException;

    EntityProxy findItemByGTIN( String gtin, String targetMarket, ChannelProxy channel,
PartyProxy informationProvider,
                             PartyProxy tradePartner, String messageType, String
publicationMarket, String context )
        throws CoreException;
}

```

There are two possible ways how an item can be stated in the REST call,

- either as Service API representation (e. g. 'MyItem'@'MySupplierCatalog', 190@45) in which case the method findItemByEntityItemReference is called
- or by a GTIN in which case the method findItemByGTIN is called

In both cases, there is all the other information that the REST call contains in order for you to find the correct item.

SDK Example

There is an SDK Example that implements the logic for items in the supplier catalogs to be found. It is in the plugin "com.heiler.ppm.customizing.gdsn.server" in the SDK examples. Here you can also find a default implementation for findItemByEntityItemReference.

2.1.11 GDSN 3.1.8 -> GDSN 3.1.9

2.1.11.1 3.1.8 → 3.1.9

GS1 Documentation:

Delta_3p1p9_v10.xlsb

GDSN_and_Shared_Code_Lists_3p1p9_8_5Jun2019.xlsx

2.1.11.2 Results:

Fields	Code list	Code list entries	Validations
2	0	69	0

2.1.11.3 Fields:

Module	Field	Change	GS1	I M	DSE	Remark
Certification information	Certification.isCertificateRequired	ADD	x			DB adjustment
Health related information	ArticleHealthCare.NutritionalScore	ADD	x			

2.1.11.4 Validations:

Validation Rules (No change in Validation Rules for this release) -> see <https://www.gs1.org/standards/gdsn/3-1-9>

2.1.11.5 Code lists:

allergenTypeCode

Code list	Entry	Change	IM	DSE
AllergenTypeCode	ABN	ADD	x	x
AllergenTypeCode	BAS	ADD	x	x
AllergenTypeCode	BLF	ADD	x	x
AllergenTypeCode	BRM	ADD	x	x
AllergenTypeCode	CHB	ADD	x	x
AllergenTypeCode	CHR	ADD	x	x
AllergenTypeCode	CLM	ADD	x	x
AllergenTypeCode	COK	ADD	x	x
AllergenTypeCode	CON	ADD	x	x

Code list	Entry	Change	IM	DSE
AllergenTypeCode	CSC	ADD	x	x
AllergenTypeCode	EEL	ADD	x	x
AllergenTypeCode	LMT	ADD	x	x
AllergenTypeCode	LSN	ADD	x	x
AllergenTypeCode	MAC	ADD	x	x
AllergenTypeCode	MAL	ADD	x	x
AllergenTypeCode	MKF	ADD	x	x
AllergenTypeCode	MM	ADD	x	x
AllergenTypeCode	MSS	ADD	x	x
AllergenTypeCode	OCT	ADD	x	x
AllergenTypeCode	ORR	ADD	x	x
AllergenTypeCode	OYS	ADD	x	x
AllergenTypeCode	PER	ADD	x	x
AllergenTypeCode	PLC	ADD	x	x
AllergenTypeCode	POM	ADD	x	x
AllergenTypeCode	PRG	ADD	x	x
AllergenTypeCode	QUA	ADD	x	x
AllergenTypeCode	ROF	ADD	x	x
AllergenTypeCode	SAR	ADD	x	x
AllergenTypeCode	SCA	ADD	x	x
AllergenTypeCode	SHK	ADD	x	x
AllergenTypeCode	SMT	ADD	x	x
AllergenTypeCode	SQU	ADD	x	x
AllergenTypeCode	STG	ADD	x	x

Code list	Entry	Change	IM	DSE
AllergenTypeCode	SU	ADD	x	x
AllergenTypeCode	TUR	ADD	x	x
AllergenTypeCode	WHK	ADD	x	x
AllergenTypeCode	WHT	ADD	x	x
AllergenTypeCode	UN	CHANGE	x	x

dietTypeCode

Code list	Entry	Change	IM	DSE
DietTypeCode	HIGH_CARB	ADD	x	x
DietTypeCode	KETO	ADD	x	x
DietTypeCode	LOW_CARB	ADD	x	x
DietTypeCode	LOW_FAT	ADD	x	x
DietTypeCode	PALEO	ADD	x	x

handlingInstructionCode

Code list	Entry	Change	IM	DSE
HandlingInstructionCode	SCB	ADD	x	x

healthClaimCode

Code list	Entry	Change	IM	DSE
HealthClaimCode	NON_GMO	ADD	x	x
HealthClaimCode	NON_TOXIC	ADD	x	x
HealthClaimCode	PHthalate_FREE	ADD	x	x
HealthClaimCode	Sulfate_FREE	ADD	x	x

nutrientTypeCode

Code list	Entry	Change	IM	DSE
NutrientTypeCode	X_FIBTPE	ADD	x	x

Code list	Entry	Change	IM	DSE
NutrientTypeCode	X_MELA	ADD	x	x
NutrientTypeCode	X_SALIC	ADD	x	x

nutritionalClaimNutrientElementCode

Code list	Entry	Change	IM	DSE
NutrientClaimNutrientElementCode	CARBOHYDRATES	ADD	x	x
NutrientClaimNutrientElementCode	CELERY	ADD	x	x
NutrientClaimNutrientElementCode	FRAGRANCE	ADD	x	x
NutrientClaimNutrientElementCode	GMO	ADD	x	x
NutrientClaimNutrientElementCode	PARABEN	ADD	x	x
NutrientClaimNutrientElementCode	PHTHALATE	ADD	x	x
NutrientClaimNutrientElementCode	SULFATE	ADD	x	x

nutritionalProgramCode

Code list	Entry	Change	IM	DSE
NutritionalProgramCode	8	ADD	x	x

platformTypeCode

Code list	Entry	Change	IM	DSE
PlatformTypeCode	99	ADD	x	x

specialtemTypeCode

Code list	Entry	Change	IM	DSE
SpecialtemTypeCode	ECO_CHEQUES	ADD	x	x
SpecialtemTypeCode	EMPLOYEE_MEAL_VOUCHERS	ADD	x	x

2.1.11.6 Units

Unit code	Label (en)	Change	IM	DSE	Remark
E20	10^6 (1000000) bits (binary digits) per second	ADD	x	x	new unit, added to GDSN units, not added to system units; is part of Enum.GDSNUnits
F79	inch of mercury	ADD	x	x	new unit, added to GDSN units, not added to system units; is part of Enum.GDSNUnits
H83	litre per kilogram	ADD	x	x	new unit, added to GDSN units, not added to system units; is part of Enum.GDSNUnits
HN	conventional millimetre of mercury	ADD	x	x	existing unit, now mapped to GDSN units; is part of Enum.GDSNUnits
N16	inch of mercury (32 °F)	ADD	x	x	new unit, added to GDSN units, not added to system units; is part of Enum.GDSNUnits
N17	inch of mercury (60 °F)	ADD	x	x	new unit, added to GDSN units, not added to system units; is part of Enum.GDSNUnits
PTN	portion	Correction	x	x	changed english label to start with a lower case character, added code to german label, because there are now two units with the same german label; changed category from miscellaneous to Count; is now part of Enum.GDSNQuantityUnits
XRO	roll (GDSN)	Correction	x	x	changed english label to start with a lower case character, changed category from scaleUnits to Count; is now part of Enum.GDSNQuantityUnits
MTS	metre per second	Correction	x	x	changed english label to start with a lower case character, changed category from scaleUnits to Count; is now part of Enum.GDSNQuantityUnits

2.1.12 GDSN 3.1.9 -> GDSN 3.1.10

2.1.12.1 3.1.9 → 3.1.10

GS1 Documentation:

Delta_3p1p10_v3.xlsb

Delta_GDSN_Validation_Rules_BMS_All_R3p1p10_i2_29Jul2019.xlsx

2.1.12.2 Results:

Fields	Code list	Code list entries	Validations
1+2	0	32	0

2.1.12.3 Fields:

Module	Field	Change	GS1	I M	DSE	Remark
Distribution details	ArticleDistributionDetail.DeliveryFrequencyCode	ADD	x			
Marketing information	ArticleMarketing.SpecialItemCode	CHANGE	x			Updated Definition
Nutritional claims	ArticleNutritionalClaim.NutritionalClaimNutrientElementCode	CHANGE	x			Updated Definition

2.1.12.4 Validations:

No rule changes.

2.1.12.5 Code lists:

dietTypeCode

Code list	Entry	Change	IM	DSE
DietTypeCode	INFANT_FORMULA	ADD	x	x
DietTypeCode	LACTASE_ENZYME	ADD	x	x
DietTypeCode	MEAL_REPLACEMENT	ADD	x	x
DietTypeCode	MOTHERS_MILK_SUBSTITUTE	ADD	x	x
DietTypeCode	NUTRITION_SUPPLEMENT	ADD	x	x
DietTypeCode	PROBIOTICS	ADD	x	x

Code list	Entry	Change	IM	DSE
DietTypeCode	TOTAL_DIET_REPLACEMENT	ADD	x	x

nutritionalClaimNutrientElementCode

Code list	Entry	Change	IM	DSE
NutrientClaimNutrientElementCode	BPA	ADD	x	x
NutrientClaimNutrientElementCode	CLONED_FOODS	ADD	x	x
NutrientClaimNutrientElementCode	FISH	ADD	x	x
NutrientClaimNutrientElementCode	FORMALDEHYDE	ADD	x	x
NutrientClaimNutrientElementCode	FORMALDEHYDE_RELEASEING_CHEMICALS	ADD	x	x
NutrientClaimNutrientElementCode	GRAINS	ADD	x	x
NutrientClaimNutrientElementCode	HYDROCARBON_PROPELLANT	ADD	x	x
NutrientClaimNutrientElementCode	HYDROQUINONE	ADD	x	x
NutrientClaimNutrientElementCode	LATEX	ADD	x	x
NutrientClaimNutrientElementCode	LEGUME_PROTEIN	ADD	x	x
NutrientClaimNutrientElementCode	NATURAL_GLUTEN	ADD	x	x
NutrientClaimNutrientElementCode	NONYLPHENOL_ETHOXYLATES	ADD	x	x
NutrientClaimNutrientElementCode	OXYBENZONE	ADD	x	x
NutrientClaimNutrientElementCode	PALM_OIL	ADD	x	x
NutrientClaimNutrientElementCode	POLYETHYLENE_MICROBEADS	ADD	x	x
NutrientClaimNutrientElementCode	PVC	ADD	x	x
NutrientClaimNutrientElementCode	PVC_WITH_PHTHALATES	ADD	x	x
NutrientClaimNutrientElementCode	PVC_WITHOUT_PHTHALATES	ADD	x	x
NutrientClaimNutrientElementCode	RETINYL_PALMITATE	ADD	x	x
NutrientClaimNutrientElementCode	SOYA	ADD	x	x
NutrientClaimNutrientElementCode	TRICLOCARBAN	ADD	x	x

Code list	Entry	Change	IM	DSE
NutrientClaimNutrientElementCode	TRICLOSAN	ADD	x	x

nutritionalClaimTypeCode

Code list	Entry	Change	IM	DSE
NutritionalClaimTypeCode	MADE_IN_FACILITY_FREE_FROM	ADD	x	x

packagingFeatureCode

Code list	Entry	Change	IM	DSE
PackagingFeatureCode	MICROWAVE_SUITABLE	ADD	x	x

packagingFunctionCode

Code list	Entry	Change	IM	DSE
PackagingFunctionCode	RECLOSABLE	ADD	x	x

2.1.13 GDSN 3.1.10 -> GDSN 3.1.11

2.1.13.1 3.1.10 → 3.1.11

GS1 Documentation:

GDSN_and_Shared_Code_Lists_3p1p11_1_Dec_11_2019.xlsx

2.1.13.2 Results:

Fields	Code list	Code list entries	Validations
0	0	22	0

2.1.13.3 Fields:

No fields changes.

2.1.13.4 Validations:

No document for validation rules. The validation delta of version 3.1.12. mentions all changes from 3.1.10 → 3.1.12.

2.1.13.5 Code lists:

dietTypeCode

Code list	Entry	Change	IM	DSE
DietTypeCode	ORGANIC	ADD	x	x
DietTypeCode	PESCATARIAN	ADD	x	x

nutrientTypeCode

Code list	Entry	Change	IM	DSE
NutrientTypeCode	X_FATRNAN	ADD	x	x
NutrientTypeCode	X_FATRNPL	ADD	x	x

nutritionalClaimNutrientElementCode

Code list	Entry	Change	IM	DSE
NutrientClaimNutrientElementCode	ALUMINIUM	ADD	x	x
NutrientClaimNutrientElementCode	AMMONIA	ADD	x	x
NutrientClaimNutrientElementCode	ANIMAL_CRUELTY	ADD	x	x
NutrientClaimNutrientElementCode	BLEACH	ADD	x	x
NutrientClaimNutrientElementCode	CORN	ADD	x	x
NutrientClaimNutrientElementCode	JUICE	ADD	x	x
NutrientClaimNutrientElementCode	MEAT	ADD	x	x
NutrientClaimNutrientElementCode	MINERAL_OIL	ADD	x	x
NutrientClaimNutrientElementCode	OPTICAL_BRIGHTENERS	ADD	x	x
NutrientClaimNutrientElementCode	PEROXIDE	ADD	x	x
NutrientClaimNutrientElementCode	PHENOXYETHANOL	ADD	x	x
NutrientClaimNutrientElementCode	SILICONE	ADD	x	x

nutritionalClaimTypeCode

Code list	Entry	Change	IM	DSE
NutritionalClaimTypeCode	EXCELLENT_SOURCE_OF	ADD	x	x
NutritionalClaimTypeCode	GOOD_SOURCE_OF	ADD	x	x
NutritionalClaimTypeCode	MADE_WITH_ORGANIC	ADD	x	x
NutritionalClaimTypeCode	PACKED_IN	ADD	x	x
NutritionalClaimTypeCode	RAISED_WITHOUT	ADD	x	x

packagingFeatureCode

Code list	Entry	Change	IM	DSE
PackagingFeatureCode	LBR	ADD	x	x

2.1.14 GDSN 3.1.11 -> GDSN 3.1.12

2.1.14.1 3.1.11 → 3.1.12

GS1 Documentation:

Delta_3p1p12_v2.xlsb

GDSN_and_Shared_Code_Lists_3p1p12_2_Jan_6_2019.xlsx

Delta_GDSN_Validation_Rules_BMS_All_R3p1p12_i5_06Feb2020.xlsx

2.1.14.2 Results:

Fields	Code list	Code list entries	Validations
5	2+1	7	3

2.1.14.3 Fields:

Module	Field	Change	GS1	I M	DSE	Remark
Cheese information	ArticleCheese.FatPercent ageInDryMatterMeasure mentPrecisionCode	ADD	x			
Health related information	ArticleHealthCare.Nutritio nalValue	ADD	x			

Module	Field	Change	GS1	I M	DSE	Remark
Health related information	ArticleHealthCare.CannabisCBDDTypeCode	ADD	x			
Health related information	ArticleHealthCare.NutritionalProgramIngredientTypeCode	ADD	x			
Health related information	ArticleHealthCareUOM.NutritionalProgramIngredientMeasurement	ADD	x			Divided into NutritionalProgramIngredientValue and NutritionalProgramIngredientUnit

2.1.14.4 Validations:

Change	Numeric Rule ID	Description	Error Message
ADD	1694	If any attributes in class microbiologicalInformation is provided, then the attribute microbiologicalOrganismCode must be provided.	If attribute XY (any attribute of class microbiologicalInformation) is provided, therefore microbiologicalOrganismCode SHALL be populated
ADD	1699	nutritionalProgramIngredientMeasurement shall only be used if nutritionalProgramIngredientTypeCode is used.	nutritionalProgramIngredientTypecode is not used.
ADD	1700	There shall be at most one iteration of nutritionalProgramIngredientMeasurement per @measurementCode	nutritionalProgramIngredientMeasurement is repeatable for @measurementUnitCode only.

2.1.14.5 Code lists:

cannabisCBDDTypeCode

Code list	Entry	Change	IM	DSE
CannabisCBDDTypeCode	FULL_SPECTRUM	ADD	x	x

Code list	Entry	Change	IM	DSE
CannabisCBDTypeCode	ISOLATE	ADD	x	x
CannabisCBDTypeCode	BROAD_SPECTRUM	ADD	x	x
CannabisCBDTypeCode	NANO_CBD	ADD	x	x
CannabisCBDTypeCode	HEMP_SEED_OIL	ADD	x	x

nutritionalProgramIngredientTypeCode

Code list	Entry	Change	IM	DSE
NutritionalProgramIngredientTypeCode	FRUITS_VEGETABLES_LEGUMES_AND_NUTS	ADD	x	x

GS1TradeItemIdentificationKeyTypeCode

Code list	Entry	Change	IM	DSE
GS1TradeItemIdentificationKeyTypeCode	ZERO_SUPPRESSED_GTIN	ADD	x	x

2.1.15 Migration Guide for IM



This migration guide describes the migration from an old GDSN datamodel 2.8 to the GDSN Major release 3 datamodel. That means if you upgrade your Product 360 version to 8.0.5 (from any older version than 7.1.08) and already had the *GDSN* and *Food and Beverage* model you have to follow the described migration steps. Please be aware that GDSN did not have defined migration pathes for all attributes, that's why it is always a good idea to contact the R&D Team before a migration is done.

The mentioned DQ rule configurations are not part of the Accelerator package. Please contact Global Customer Support if you need them.

2.1.15.1 General changes

Unit System

The unit system "1WS" will be deleted. Further on the unit system "GDSN" should be used for all unit fields.

Unit

The standard unit fields will be migrated according to the MjR 3 migration guide.

The previously used unit "Fluid ounce" with code "FZ" has to be transferred manually to a valid unit. The value is only allowed for the field 'Product yield UOM' which has a data quality configuration to find items with this value in use. Please ensure that no unit field is still using this unit.

In case this pre-migration step is not executed then the unit will automatically merged to "fluid ounce (UK)" with the code "OZI".

All unit fields got a unit filter which means, only values which are allowed for the specific field will be displayed. In case you did use an invalid unit, you won't see the maintained unit after migration.

Repository

'Replaced by GTIN' and 'Replaces GTIN'

Those fields must be exactly 14 digits long, otherwise an error will occur.

Pricing on product

The pricing on product field became a three-state boolean ('true', 'false', null).

2.1.15.2 GDSN Core Attributes

Product Type

Assigned to the field Product type (GDSNExtension.ProductType) is the valid values list GDSN trade item unit descriptor values. In this list, three entries were entirely removed. These are Prepack (PP), Multipack (MP) and Prepack assortment (PA). Please check if you used one of these values and if so please adjust your data.

GTIN variation registry

This field is deprecated and has been removed.

If you want to check if you have items which have a GTIN variation registry populated with "true", you can create a DQ rule configuration with the following configuration:

Name*:
Checks if the field GTIN variation registry (USA) does not contain the value 'true'

Description:
Checks if the field GTIN variation registry (USA) does not contain the value 'true'.

Rule name*:
Check_ConditionalNotEqual

Data type:

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField1	string(512)	"1"
inConditionValue1	string(512)	"1"
inConditionField2	string(512)	"1"
inConditionValue2	string(512)	"1"
inCheckField	string(512)	GTIN variation registry (USA)
inCheckValue	string(512)	"true"
inStatusMessage	string(50)	

Data type:

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

This DQ rule configuration will fail for every item which has a GTIN variation registry set to "true".

EANUCC type respectively TradeItemIdentificationKey

All values from the valid values list were entirely removed by GDSN. There is no migration path given by GDSN.

Here is our migration path:

Preparation:

Run DQ configurations:

- "Check 'Target market' of EANUCC is not equal 'Worldwide'" which will find out data with target market = "Worldwide"
- All DQ configuration in the "'EAN UCC type' to 'TradeItemIdentificationKey'" category. Those will find data that cannot be converted because of duplicate codes, example: both types UD and UE would create same entry with new type GTIN_12

Adjust EANUCC type data for items that didn't pass the DQ checks successfully:

- Delete all entries with target market = "Worldwide" or change the target market to a valid value, e.g. "US"
- Adjust the data, so that the entries can be converted, e.g. remove entries with types that would create duplicate entries

Migration

We'll convert the following types. All existing entries that would create duplicate entries during conversion will be skipped. All entries will be skipped that got the target market "Worldwide".

New code	old code	old label
GTIN_8	EO	European Article Number 8 (EAN)
GTIN_12	UD	UPC/EAN Consumer Package Code
	UE	UPC/EAN Module Code
	UG	Drug UPC Consumer Package Code
	UI	UPC Consumer Package Code
	UP	UCC-12
GTIN_13	EN	European Article Number (EAN)
GTIN_14	U2	UPC Shipping Container Code
	UA	UPC/EAN Case Code
	UH	Drug UPC Shipping Container Code
	UK	UPC/EAN Shipping Container Code
	UN	UPC Case Code Number

Packaging type code

The packaging specific information was global and became target market specific now. The list of packaging types was completely changed, it was reduced from about 140 to 40 codes. In addition, some of the types are represented now by a packaging type in combination with a value of another field.

There's a migration path for most packaging types given by GDSN, see "1WorldSyncDeltaDoc_IM7.0_forMjR3_v10.xlsx".

- Most packaging type codes can be uniquely migrated to a new code.
- Some codes are not mapped to a new code, you have to find a suitable replacement attribute/value combination.
- Some codes will be migrated to the new code and a value in another field, e.g. "Packaging feature code"

Prepare migration

You need to find out if there's data that cannot be migrated. Therefore you should run the following DQ check:

- "Check 'Packaging type' is not equal 'Splash blend'"

Migration

We'll consider all packaging type codes that are listed in "1WorldSyncDeltaDoc_IM7.0_forMjR3_v10.xlsx". All other packaging type codes won't be migrated.

- We'll migrate the codes as defined in the delta document. Most packaging type codes can be uniquely migrated, but we don't create duplicate entries if two codes are mapped to the same new code.
- Codes that are not mapped to a new code but got a "No Migration Done" comment, will be migrated to "No code", so that you can adjust those entries after the automatic migration has been done.
- Some codes will be migrated to the new code and a value in another field, e.g. "Packaging feature code", as defined in the delta document.
- We'll set the target market to "USA" for all packaging type entries
- Furthermore, we'll migrate some additional packaging type codes as listed below:

Old packaging type	Migrated packaging type
GB (Bottle, gas)	BO (Bottle)
1A (Drum, steel)	BA (Barrel)
4H (Box, plastic)	BX (Box)
BIB (Bottle in box)	PUG (Packed, unspecified)
CI (Canister)	PUG (Packed, unspecified)
CN (Container not otherwise specified as transport equipment)	PUG (Packed, unspecified)
CTN (Carton)	PUG (Packed, unspecified)
NA (Not available)	PUG (Packed, unspecified)

Limitations:

- The old packaging code "Splash blend" (AAB) will be migrated to "Package, unspecified", but we won't set TradeltemDescription or AdditionalTradeltemDescription to 'Splash Blend'.
- We don't create duplicate entries if two codes are mapped to the same new code. In such cases, only one entry gets migrated.
- The GDSN migration path defines an update of the product type to "Transport load" for some codes. We'll migrate the packaging code, but we won't update the product type value.

Post-migration

After automatic data migration you probably need to adjust data of items you found by DQ checks.

You should check all items that got a "Package, unspecified" or "No code" packaging type code.

StartAvailabilityDate and EndAvailabilityDate

The fields GDSNTargetMarketExtension.StartAvailabilityDate, GDSNTargetMarketExtension.EndAvailabilityDate, GDSNCustomerSpecificExtension.StartAvailabilityDate and GDSNCustomerSpecificExtension.EndAvailabilityDate have been moved to ArticleDeliveryPurchasing.StartAvailabilityDate and ArticleDeliveryPurchasing.EndAvailabilityDate.

The migration works as follows: The values from GDSNTargetMarketExtension.StartAvailabilityDate and GDSNTargetMarketExtension.EndAvailabilityDate will be moved to ArticleDeliveryPurchasing.StartAvailabilityDate and ArticleDeliveryPurchasing.EndAvailabilityDate. If the delivery purchasing information entry for the target market is available it will be created. If startAvailability and endAvailability date already exist for delivery purchasing information, we keep these values and the dates from target market specific data will be discarded. All dates from the customer specific data will be discarded.

2.1.15.3 Food and Beverage

Food and Beverage has always had a logical key "TargetMarket". However, this logical key was disabled in the standard and the default value was WORLD. In GDSN 3.0 the "TargetMarket" has been activated and 'WORLD' is not part of the valid values list. Therefore we decided to change all data sets with "TargetMarket" = 'WORLD' to "TargetMarket" = 'US' during migration. Data sets with other target markets are not affected and keep their old value.

Soft deleted data won't be migrated.

Product Yield

So far the value of "Product yield" has been formatted with a scale of two. With the new release the scale got changed to six. Before, in case you did write a value with a higher precision, the value was formatted to a scale of two. After the migration you are able to see and export those additional scale.

Example

Old data:

Input value of 'Product yield'	Display value of 'Product yield'
1.22	1.22
1.123	1.12
1.3294	1.33

Migrated data:

Display value of 'Product yield'
1.220000
1.123000
1.329400

Product Yield UOM

The unit "Fluid ounce" with the code "FZ" will be removed. If you want to keep the data, you need to change the UOM to a valid value regarding GDSN 3.0. Otherwise you won't see the UOM after the migration anymore.

Diet information

Regarding **language-specific data**, there is a huge change. In the old model one description could be maintained per combination of diet certification agency and diet certification number. In the new model, there is only one description per target market. This means n descriptions have to be reduced to one. We cannot decide for our customers which descriptions to keep and which ones to discard. Therefore we won't migrate any Diet descriptions.

To handle this situation you could for example export all descriptions of the old system, decide which ones to keep and add them in the migrated system.

In GDSN 3.0 the **diet type** "ORGANIC" has been removed completely. There is a DQ rule configuration ("Check 'Diet type' is not equal 'Organic'") to find all diets with this diet type, so you can run it **before the migration** and adjust your data accordingly.

We don't migrate **diet subtype** because there's no migration path given by GDSN, and the old and the new valid values list have no values in common. No entries will be lost, only the "Diet subtype" field value will be empty. If you want to maintain new values on the basis of the old data you can do an export of the diet information data before the migration and decide which value of the new valid value list to maintain.

Since the data model changed a lot, there are some data sets which cause problems during the migration. They have different diet certification agencies and diet certification numbers but the same diet type.

The value of the field `ArticleDietRelatedInformation.DietCertificatonAgency` will be mapped to the field `ThirdParty.Name`. So for each single **certification agency** specified in diet related information a new "third party" will be created with the given name. The third parties are referenced by the `Certification` which is referenced again by the `ArticleDietList`. So the full path to the diet **certification agency** is now:

```
ArticleDietList.Certification->Certification.Organisation->ThirdParty.Name
```

The value of the field `ArticleDietRelatedInformation.DietCertificatonNumber` will be mapped to the field `Certificate.Value`. So the full path to the diet **certification number** is now:

```
ArticleDietList.Certification->Certificate.Value
```

Nutrient information and Serving instance

In our GDSN 2.8 Food and Beverage data model, there used to be a logical key "PartyID". This logical key will no longer be available in the MjR3 model. If you activated this key, be careful with the migration. You have to take care of possible doubled entries yourself and remove them before the migration. All your data sets need

to have a unique value combination for the keys available in MjR3. See the repository to have a look at the new datamodel.

In the MjR3 datamodel, a new logical key "Nutrient basis quantity type" is introduced. Its values are BY_MEASURE or BY_SERVING. They replace the serving instance of the PIM GDSN 2.8 model. Serving instance "per 100g" will be migrated to BY_MEASURE, "per serving" will be migrated to BY_SERVING. Data sets with serving instance "per package" won't be migrated. There is the DQ rule configuration "Check 'Serving instance' is not equal 'per package' " to find all nutrient information with serving instance "per package", so you can adjust your data **before the migration** if you want to keep it.

Serving instances not used by nutrients will not be migrated.

The new entities "Nutrient basis quantity" and "Nutrient list quantity" have the logical key UOMType. Nutrients and ServingInstance in the old model didn't have an equivalent logical key or field. We decided to insert the value "Imperial" during the migration for all data sets. You might want to have a look at this after the migration.

The former household serving size could be maintained for serving instance per 100g, per serving and per package individually. Since this field has been renamed to serving size description in GDSN 3, we only migrate the value of the household serving size belonging to per serving. Household serving sizes for serving instances "100 g" and "per package" won't be migrated. You can use an export before the migration to get these data and to decide which values you really want to keep and maybe change it after the migration.

Preparation information

The datamodel changed regarding the convenience level. There used to be one convenience level per product yield type. This is no longer the case. This means some data sets have to be merged. Our migration will take the lowest convenience level. You might want to have a look at your data **before** the migration. You can use an export to see where you have different convenience levels for the same target market and preparation type and decide later which value to use and adjust the data after the migration manually. Alternatively you can remove the values you don't want before the migration.

Similar change happened to preparation instructions. In our GDSN 2.8 compliant datamodel, you could maintain one preparation instruction per product yield type. This is no longer the case. Depending on your database, either the first preparation instruction regarding alphabetical sorting (MS SQL) will be kept, or the first preparation instruction created (Oracle) will be kept. You might want to have a look at your data **before** the migration and either delete the values you don't want to keep or do an export with the data to decide later which values to keep and if necessary change them after the migration manually.

The UOM type of product yield UOM will be set to 'IMPERIAL'.

Example:

Old data:

Item no.	Preparation type	Product yield type	Convenience level	Preparation instruction (German)	Creation Date of preparation instruction (German)	Preparation instruction (English)	Creation Date of preparation instruction (German)
Item 1	Baking	After cooking	3.00	Zuerst	01.01.2000	First ...	10.10.2000

Item no.	Preparation type	Product yield type	Convenience level	Preparation instruction (German)	Creation Date of preparation instruction (German)	Preparation instruction (English)	Creation Date of preparation instruction (German)
Item 1	Baking	Drained weight	13.00	Nehmen ...	10.10.2000	Take	01.01.2000

Migrated data MS SQL:

Item no.	Preparation type	Product yield type	Convenience level	Preparation instruction (German)	Preparation instruction (English)
Item 1	Baking	n/a	3.00	Nehmen ...	First ...

Migrated data Oracle:

Item no.	Preparation type	Product yield type	Convenience level	Preparation instruction (German)	Preparation instruction (English)
Item 1	Baking	n/a	3.00	Zuerst ...	Take ...

Certification information

The value of the old field `ArticleCertificationInformation.CertificatonAgency` will be mapped to the field `ThirdParty.Name`. So for each single **certification agency** specified in the certification information a new "third party" will be created (if not exists yet) with the given name. The third parties are referenced by the new root-entity `Certification` which is referenced again by the new module `ArticleCertifications`. So the full path to the **certification agency** is now:

`ArticleCertifications.Certification -> Certification.Organisation -> ThirdParty.Name`

The value of the old field `ArticleCertificationInformation.CertificatonStandard` will be mapped to the field `Certification.Standard` of the new root-entity `Certification` which is referenced by the new module `ArticleCertifications`. So the full path to the **certification standard** is now:

`ArticleCertifications.Certification -> Certification.Standard`

The value of the old language-specific field `ArticleCertificationInformationLang.CertificatonValue` will be mapped to the field `Certificate.Value` of the new root-entity `Certification` which is referenced by the new module `ArticleCertifications`. Since the field `Certificate.Value` is not language-specific - only the values for the language **US** will be migrated. So the full path to the **certification value** is now:

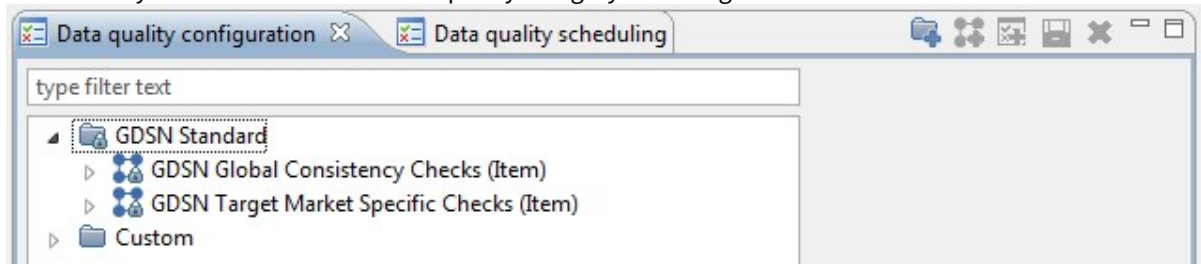
`ArticleCertifications.Certification -> Certificate.Value`

2.1.15.4 Merge of migration Data Quality configurations

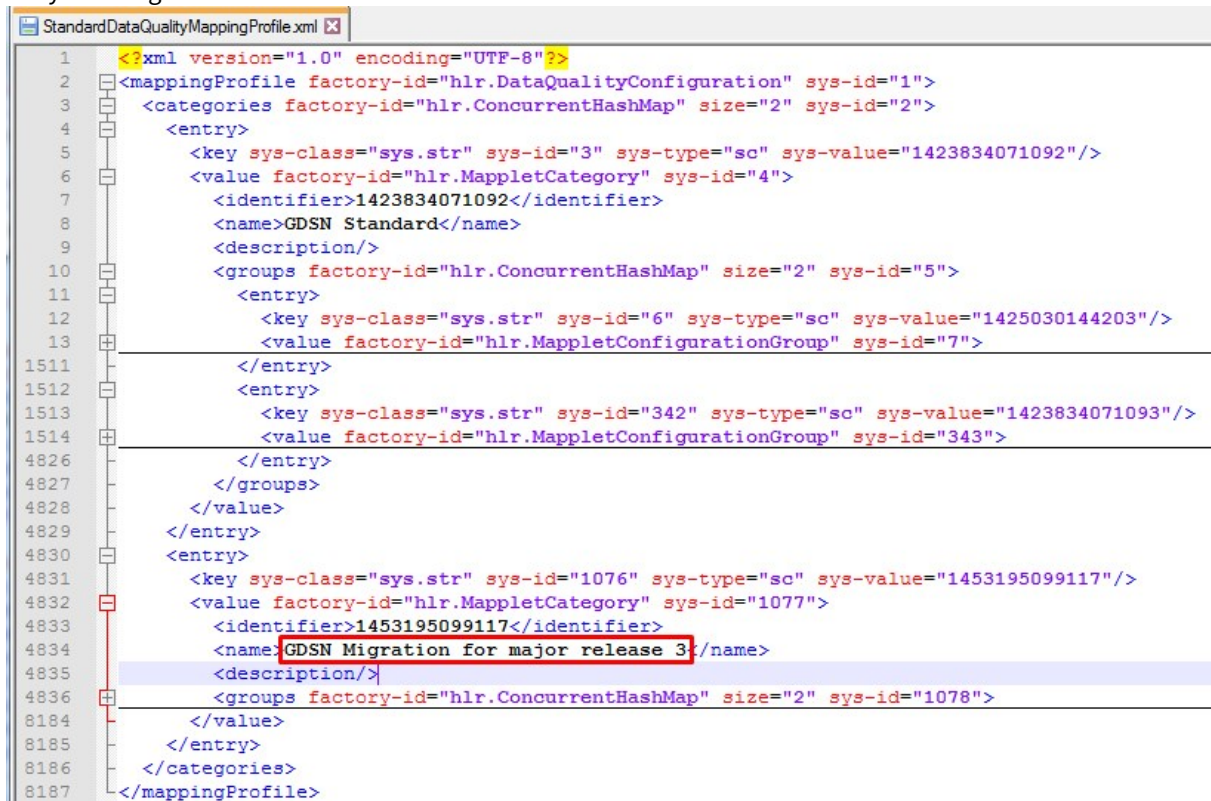
As mentioned above, if you need the mentioned Data quality rule configuration which can help you to migrate your data, please contact Global Customer Support.

Instead of using the whole delivered standard GDSN Data Quality package for the Migration of PIM 7.1.05.xx, it is possible to merge the needed migration data quality configuration manually as described below.

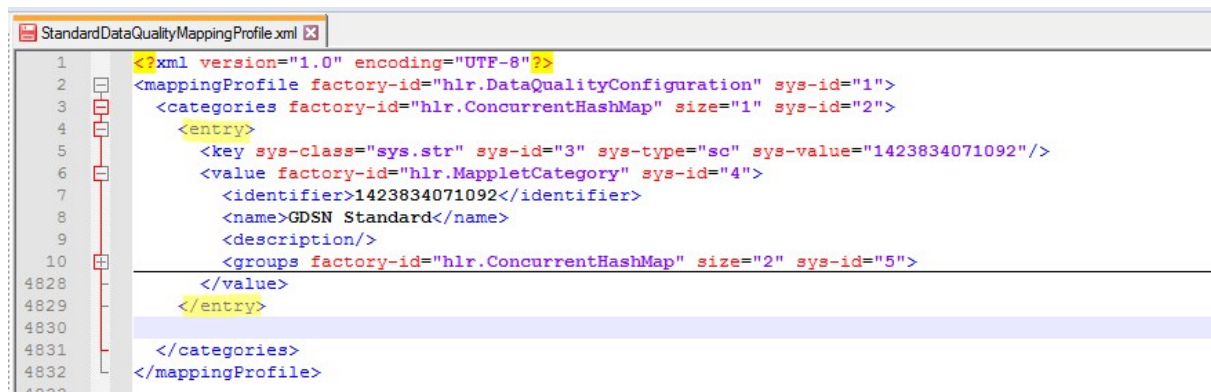
1. Ensure that your 'GDSN Standard' data quality category is looking like shown beneath.



2. Open the StandardDataQualityMappingProfile.xml of the delivered GDSN package (RuleConfigurationPackageGDSN/dataquality/config)
3. Search for 'GDSN Migration for major release 3' to find the appropriate entry and copy the whole entry- XML tag.



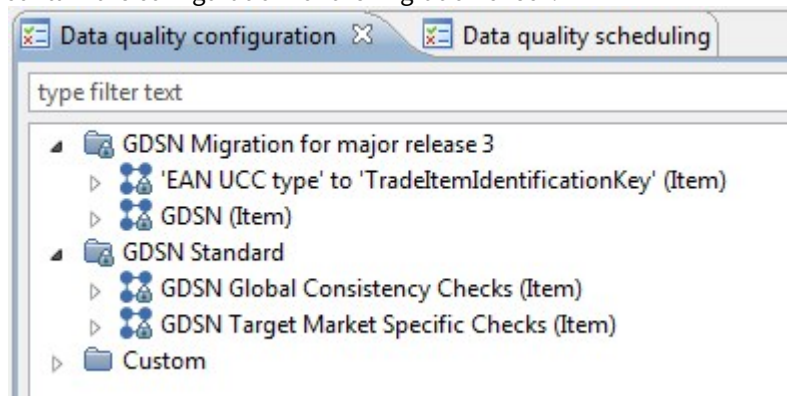
4. Navigate to your Data Quality folder, which is defined in the server.properties file as dataquality.root.local
5. Open the StandardDataQualityMappingProfile.xml in the config folder.
6. Paste the entry-tag from above after the existing entry tag in this file. This would be line number 4830 in the image below.



The resulting StandardDataQualityMappingProfile.xml should look like below



- After those changes you need to restart your PIM server. After restart the Data quality configuration view will contain the configuration for the migration check.



2.1.15.5 Missing Data Quality rules

The following rules are missing and can be migrated at a later time:

Consumer unit

If 'Is consumer unit' (IsConsumerUnit) is equal to true and 'Data carrier type' (dataCarrierTypeCode) is equal to (EAN_13, UPC_A or UPC_E) then the first digit of the GTIN must equal 0.

If 'Is consumer unit' (IsConsumerUnit) is equal to true and the 'Target market' is US, the type of bar code 'Data carrier type' (dataCarrierTypeCode) is mandatory.

Data carrier type

If 'Is consumer unit' (IsConsumerUnit) is equal to 'true' and 'Data carrier type' (dataCarrierTypeCode) is equal to 'EAN_8' then the first six digits of the TradeItem/GTIN must equal '000000'.

This error does not only occur with this data carrier type, the same behavior can occur with other data carrier types.

Packaging weight

The total of values in 'Packaging material composition quantity' may not exceed the value in 'Packaging weight' for the same item.

Product type

If the target market does not equal AU (Australia) or NZ (New Zealand) it is mandatory to specify the pallet type (Platform type code) when product type is equal to PALLET.

2.1.16 GDSN Migration Guide for version 3.1.3

This migration guide is pointing out the changes which have been done for Product 360 v.8.0.05.05, 8.0.6.03 and higher in order to support the GDSN version 3.1.3. Furthermore the guideline is based on previous 8.0.5.xx and 8.0.6.xx Product 360 versions.

2.1.16.1 Overview

The migration guideline is going through all changes which have been made for the GDSN version 3.1.3 in the following chapters. In detail it's presenting following topics:

- Data cleansing which has to be done before executing the Product 360 update
- Subentity changes grouped by GDSN modules
- General data model changes which have been done in the repository
- Data quality rule and configuration changes in the GDSN Accelerator package
- Additional changes which have been done independently of GDSN version 3.1.3

2.1.16.2 Prerequisite for the Migration

In order to not destroy your data in your Product 360 system, some data quality checks must be executed before the migration can be started. This is necessary to be done because some values will be deleted during the update.

FoodAndBeveragePropertiesInformationModule

MicrobiologicalInformation

IM

Please make sure that all data quality rule configurations in the category "Microbiological information" are executed and passed in order to have migration-ready data. If one of the values does not have a corresponding UOM, another UOM could be assigned or the value gets lost during the migration.

DSE

Please ensure that all microbiological values have a corresponding UOM defined. If one of the values does not have a corresponding UOM, another UOM could be assigned or the value gets lost during the migration.

To check your data please create following data quality rule configurations and execute them. All microbiological information data entries need to pass those configurations in order to be ready for migration.

1. Create a data quality rule configuration "Check items with 'Organism maximum value' have a corresponding UOM populated" with following settings:
 - Based on data quality rule: Check_IfNotEmptyConditionNotEmpty
 - Using data type: Microbiological information
 - Mapping for input port "inConditionField": Organism maximum value
 - Mapping for input port "inCheckField": Organism maximum value UOM

The screenshot shows the 'Data quality configuration' window. On the left, a tree view shows the hierarchy: GDSN3 > MicrobiologicalMigration > Microbiological information (Item) > Check that items with 'Organism maximum value' have a corresponding UOM populated. The right pane shows the configuration for this rule.

Name: Check that items with 'Organism maximum value' have a corresponding UOM populated ☐ Hidden

Description: Checks that [inCheckField] is not empty, if and only if the condition [inConditionField] is not empty. Returns: [outStatusMessage],[outStatusCode] -> (no Error, OK): If [inConditionField] is empty or both, [inConditionField] and [inCheckField], are not empty. In case of a failure there are two possible status message texts: [inStatusMessage] contains an GDSN error code which refers to a reference table entry, in this case [outStatusMessage] should contain the specific message.

Rule name: Check_IfNotEmptyConditionNotEmpty

Data type: Microbiological information

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Organism maximum value
inCheckField	string(512)	Organism maximum value UOM
inStatusMessage	string(50)	

Data type: Item

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

Data processor: [Dropdown menu]

2. Create a data quality rule configuration "Check items with 'Organism reference value' have a corresponding UOM populated" with following settings:

- Based on data quality rule: Check_IfNotEmptyConditionNotEmpty
- Using data type: Microbiological information
- Mapping for input port "inConditionalField": Organism reference value
- Mapping for input port "inCheckField": Organism reference value UOM

The screenshot shows the 'Data quality configuration' window. On the left, a tree view shows the hierarchy: GDSN3 > MicrobiologicalMigration > Microbiological information (Item) > Check that items with 'Organism reference value'. The main area on the right contains the following configuration details:

Name: Check that items with 'Organism reference value' have a corresponding UOM populated ☐ Hidden

Description: Checks that [inCheckField] is not empty, if and only if the condition [inConditionField] is not empty. Returns: [outStatusMessage],[outStatusCode] -> (no Error, OK): If [inConditionField] is empty or both, [inConditionField] and [inCheckField], are not empty. In case of a failure there are two possible status message texts: [inStatusMessage] contains an GDSN error code which refers to an reference table entry, in this case [outStatusMessage] should contain the specific message.

Rule name: Check_IfNotEmptyConditionNotEmpty

Data type: Microbiological information

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Organism reference value
inCheckField	string(512)	Organism reference value UOM
inStatusMessage	string(50)	

Data type: Item

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

Data preprocessor:

3. Create a data quality rule configuration "Check items with 'Organism warning value' have a corresponding UOM populated" with following settings:

- Based on data quality rule: Check_IfNotEmptyConditionNotEmpty
- Using data type: Microbiological information
- Mapping for input port "inConditionalField": Organism warning value

- Mapping for input port "inCheckField": Organism warning value UOM

The screenshot shows the 'Data quality configuration' window. On the left, a tree view shows the hierarchy: GDSN3 > MicrobiologicalMigration > Microbiological information (Item) > Check that items with 'Organism warning'. The selected rule is 'Check that items with 'Organism warning value' have a corresponding UOM populated'. The right pane shows the rule details:

Name: Check that items with 'Organism warning value' have a corresponding UOM populated ☐ Hidden

Description: Checks that [inCheckField] is not empty, if and only if the condition [inConditionField] is not empty. Returns: [outStatusMessage],[outStatusCode] -> (no Error, OK): If [inConditionField] is empty or both, [inConditionField] and [inCheckField], are not empty. In case of a failure there are two possible status message texts: [inStatusMessage] contains an GDSN error code which refers to an reference table entry, in this case [outStatusMessage] should contain the specific message.

Rule name: Check_IfNotEmptyConditionNotEmpty

Data type: Microbiological information

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Organism warning value
inCheckField	string(512)	Organism warning value UOM
inStatusMessage	string(50)	

Data type: Item

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

Data preprocessor: [Dropdown menu]

2.1.16.3 Migration

Changes in GDSN modules

MarketingInformationModule

A new sub-entity "ArticleMarketing" has been introduced for the "MarketingInformationModule". The new sub-entity comes with a third-level sub-entity "ArticleMarketingLang" which is used for language-dependent data. Already existing fields of the "MarketingInformationModule" were moved from sub-entity "GDSNTargetMarketExtension".

Moved fields:

- Coupon family code
 - old identifier: GDSNTargetMarketExtension.CouponFamilyCode
 - new identifier: ArticleMarketing.CouponFamilyCode
- Special item code
 - old identifier: GDSNTargetMarketExtension.SpecialItemCode
 - new identifier: ArticleMarketing.SpecialItemCode
- Marketing message
 - old identifier: GDSNTargetMarketExtensionLang.MarketingMessage
 - new identifier: ArticleMarketingLang.MarketingMessage

Added fields:

- Sequence (identifier: ArticleMarketingLang.Sequence)

For this change following adjustments have been made:

- Introduced a new view called "Marketing information" for Product 360 Desktop client

	Target market	Coupon family code	Special item code
1	Germany	003	Collateral item; Gift with purchase

	Language	Sequence	Marketing message
1	German	2	Another test marketing message in german
2	German	1	Test Marketing message in german
3	English	1	Test Marketing message in english

- Introduced a new view called "Marketing information" for Product 360 Web client

Target market: Germany

Coupon family code: 003

Special item code: Collateral item, Gift with purchase

Target market	Language	Sequence	Marketing message
Germany	German	2	Another test marketing message in german
Germany	English	1	Test Marketing message in english
Germany	German	1	Test Marketing message in german

- A logical key "Sequence" was added to the entity "ArticleMarketingLang" to be able to maintain more than one marketing message per language (see also [Product 360 data model changes -> ArticleDomainLangType](#)(see page 318))
- A new import function "sequence" has been introduced which generates a sequence number to a logical key or a field.
- Existing data quality rule configurations got adjusted accordingly.
- Export format templates to export the data of this module have been adjusted accordingly.



All values of the moved fields will be automatically migrated to the new sub-entity. There are no manual steps or any migration paths to consider.



The new repository sub-entity "ArticleMarketing" is disabled by default (as all GDSN entities) and will be enabled during the server start.

Marketing message

The field "Marketing message" (in GDSN "tradeItemMarketingMessage") was moved from third-level sub-entity "GDSNTargetMarketExtensionLang" to the newly introduced third-level sub-entity "ArticleMarketingLang". In addition a new field "Sequence" has been added to be able to maintain multiple marketing messages for one language.

Note: The "sequence" represents a not negative number.

SpecialItemCode

This field has been moved from "GDSNTargetMarketExtension" to the new sub-entity "ArticleMarketing".

CouponFamilyCode

This field has been moved from sub-entity "GDSNTargetMarketExtension" to the new sub-entity "ArticleMarketing".

In addition, this field became a multi value field. This means the user can maintain more than one value. The export format templates to export the data of this field has been adjusted accordingly.

Data quality (IM): The data quality rule configurations "Check 'Coupon family code' has max length of 3 characters (TM: US)" and "Check 'Coupon family code' has min length of 3 characters (TM: US)" have been adjusted accordingly. Find detailed information in chapter [Updated data quality rule configurations \(IM\) -> Target market specific checks / Marketing information](#)(see page 321).

Target market specific GDSN attributes

GDSN short description (GDSNTargetMarketExtensionLang.ShortDescription)

The maximum length of this field has been reduced to 35 characters.

Technical information: The maximum length of that field has been adjusted in the repository.

Changes in Food and Beverage modules

DairyFishMeatPoultryItemModule

Cheese maturation period description (ArticleCheeseLang.CheeseMaturationPeriodDescription)

From now on multiple language-dependent data of "Cheese maturation period description" can be exported. The field did already support multiple languages in Product 360, so no changes have been applied to this field. Export format templates to export the data of this module have been adjusted accordingly.

NutritionalInformationModule

Daily value intake reference (ArticleNutrientLang.DailyValueIntakeReference)

The maximum length of this field was changed from 70 to 500 characters.

Technical information: The field type was changed to ArticleDomainLangType.Std_Text1000_01. All values will be migrated automatically to the new database column, there are no migration paths to consider.

Serving size (ArticleNutrientBasisQuantity.ServingSize)



To retain the usability in this area it has been decided **not** to enable multiple serving sizes and their respective units by default. When desired this can be achieved by enabling the reserved logical keys in the data model (unit and text), deactivating today's serving size field, enhancing the enumeration which qualifies the quantity contained field and adjusting the export.

FoodAndBeveragePropertiesInformationModule

Microbiological information

From now on all three values can be maintained per unit. That means for example that you can define an "organism maximum value" for a weight as well as for a volume. See the screenshot below for a better understanding.

The screenshot shows two windows. The top window, 'Microbiological information (1)', has a table with columns: Target market, Organism code, Organism maximum value (metric, milligram), and Organism maximum value (metric, millilitre). It contains one row: USA, Fungi, 5.000000, 10.000000. The bottom window, 'Microbiological detail information (2)', has a table with columns: UOM type, Unit, Organism maximum value, Organism reference value, and Organism warning value. It contains two rows: 1 metric, milligram, 5.000000, 2.00, 4.00; and 2 metric, millilitre, 10.000000, 4.00, 8.00.

Target market	Organism code	Organism maximum value (metric, milligram)	Organism maximum value (metric, millilitre)
USA	Fungi	5.000000	10.000000

UOM type	Unit	Organism maximum value	Organism reference value	Organism warning value
1 metric	milligram	5.000000	2.00	4.00
2 metric	millilitre	10.000000	4.00	8.00

Affected fields:

- OrganismWarningValue
 - identifier: ArticleMicrobiologicsUOM.OrganismMaximumValue
- OrganismReferenceValue
 - identifier: ArticleMicrobiologicsUOM.OrganismReferenceValue
- OrganismMaximumValue
 - identifier: ArticleMicrobiologicsUOM.OrganismMaximumValue

For this change following adjustments have been made:

- Introduced a new view for Product 360 Desktop client called "Microbiological detail information" (see screenshot above)
- Adjusted the view "Microbiological information" of Product 360 Web client

The screenshot shows the Product 360 Web client interface with tabs: Fish meat poultry content, Diet related information, Microbiological information, Nutrients, Nutrient information, Nutritional claims, Preparation serving information, Product yield, and Settings. The 'Microbiological information' tab is active, showing a table with columns: Target market, Organism code, UOM type, Unit, Organism maximum value, Organism reference value, and Organism warning value. It contains two rows: Germany, Milk acid, metric, decigram, 5.000000, 10.12, 6.00; and Germany, Fungi.

Target market	Organism code	UOM type	Unit	Organism maximum value	Organism reference value	Organism warning value
Germany	Milk acid	metric	decigram	5.000000	10.12	6.00
Germany	Fungi					

- Introduced a logical key as UOM value (identifier "ArticleMicrobiologicsUOM.LK.Unit")
- Removed the UOM fields with identifier "ArticleMicrobiologicsUOM.OrganismMaximumValueUOM", "ArticleMicrobiologicsUOM.OrganismReferenceValueUOM", "ArticleMicrobiologicsUOM.OrganismWarningValueUOM"

Please note that it's possible that you have more entries after the migration than before due to the fact that the unit became a logical key. The three old unit fields and their data have been removed.

Find detailed technical information in chapter [Product 360 data model changes -> ArticleDomainUOMType](#)(see page 316).



Please verify that all maintained value fields have an according UOM value maintained before the migration.



All values of the moved fields will be automatically migrated to the new sub-entity. There are no manual steps or any migration paths to consider.



Community Profiles

Certain community profiles (e.g FMCG) have a limitation of possible UOM values which are defined in the "Enum.GDSNMicrobiologicalUnits" enumeration. If you want to limit to the predefined values, you have to select this enumeration for the corresponding UOM field (ArticleMicrobiologicsUOM.Unit) in the repository.

Physiochemical information

The field "Physiochemical characteristic value UOM" (identifier: ArticlePhysioChemicalUOM.PhysiochemicalCharacteristicValueUOM) is no longer mandatory.

IM:

The mandatory check was done by the data quality rule configuration "Check items with 'Physiochemical characteristic value' have a corresponding UOM populated" which has been removed.

Export format template has been adjusted accordingly.

DSE:

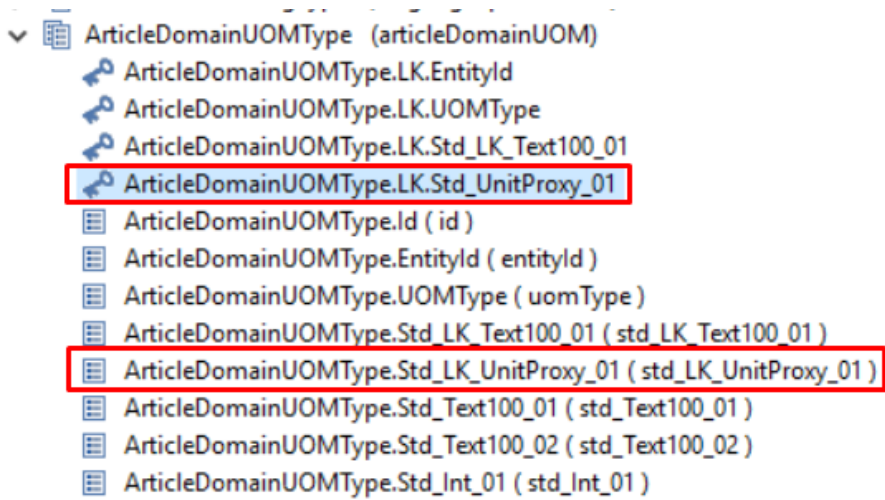
Export format template has been adjusted accordingly.

2.1.16.4 Product 360 data model changes

Some data model changes became necessary in the types area of the repository in order to support the GDSN version 3.1.3.

ArticleDomainUOMType

A new logical key type "ArticleDomainUOMType.LK.Std_UnitProxy_01" has been introduced. The standard custom entities have been adjusted accordingly.



All existing custom entities which are based on this entity type must be manually extended by corresponding logical key and field.

In case you need to deactivate this additional key use the default value 1 and do not maintain an enumeration:

The top screenshot shows the Properties window for a logical key. The 'Logical Key Type' property is highlighted, with a value of 'ArticleDomainUOMType.LK.Std_UnitProxy_01'. The 'Name' property is set to '%logical-key.ArticleDairyFishMeatPoultryUOM.LK.Std_UnitProxy_01'.

The bottom screenshot shows the Properties window for a field. The 'Field Type' property is highlighted, with a value of 'ArticleDomainUOMType.Std_LK_UnitProxy_01 (std_LK_UnitProxy_01)'. The 'Value' property is set to '1'.

ArticleDomainLangType

A new logical key type "ArticleDomainLangType.LK.Std_LK_Int_01" has been introduced. The standard custom entities have been adjusted accordingly.

- ▼ ArticleDomainLangType (languageSpecificData)
 - ArticleDomainLangType.LK.EntityId
 - ArticleDomainLangType.LK.Language
 - ArticleDomainLangType.LK.Std_LK_Int_01
 - ArticleDomainLangType.Id (id)
 - ArticleDomainLangType.EntityId (entityId)
 - ArticleDomainLangType.Language (language)
 - ArticleDomainLangType.Std_LK_Int_01 (std_LK_Int_01)
 - ArticleDomainLangType.Std_Text100_01 (std_Text100_01)
 - ArticleDomainLangType.Std_Text100_02 (std_Text100_02)

⚠ All existing custom entities which are based on this entity type must be manually extended by corresponding logical key and field.

2.1.16.5 Data quality rule and configuration changes

Some changes have been made for the data quality component. This contains new, removed and updated data quality rule configurations as well as new and updated GDSN data quality rules.

New data quality rule configurations

Global consistency checks

"Check 'Effective date' is not empty"

The new data quality rule configuration "Check 'Effective date' is not empty" was added for IM and DSE. This rule configuration ensures that the field "Effective date" is set.

Name*:
☐ Hidden

Description:

Checks if 'Effective date' is not empty.

Rule name*:

Data type:

Input port	Data type	Field
Field_Value	string(4096)	Effective date

Data type:

Output port	Data type	Field
Status_Message	string(1024)	Message
Status_Code	string(10)	Status

Data preprocessor:

Ingredient information (New)

"Check 'Ingredient Name (en)' is not empty"

The new data quality rule configuration "Check 'Ingredient Name (en)' is not empty" was added to the new group "Ingredient information" for IM and DSE. This rule configuration ensures that an English "ingredient

name" is set when a "sequence" was defined. In case you want to ensure this for another language, please copy the configuration in your custom data quality configuration category and adjust it for your language.

Limitation: The sequence is optional in the GDSN data model but it is mandatory in Product 360.

Name*: ☐ Hidden

Description:

Rule name*:

Data type:

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Sequence
inCheckField	string(512)	Ingredient (Sequence selectable, Tar...
inStatusMessage	string(50)	"GDSN1177"

Data type:

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

Removed data quality rule configurations (IM)

Target market specific checks

"If 'Pricing on product is not empty, then 'Is consumer unit' must equal true (TM: US)"

The data quality rule configuration "If 'Pricing on product is not empty, then 'Is consumer unit' must equal true (TM: US)" for IM has been removed because it's no longer valid due to the GDSN changes.

Microbiological information

All data quality rule configurations in the group "Microbiological information" has been removed because they are no longer needed after the data model change. It contained following data quality rule configurations:

- "Check items with 'Organism maximum value' have a corresponding UOM populated"
- "Check items with 'Organism reference value' have a corresponding UOM populated"
- "Check items with 'Organism warning value' have a corresponding UOM populated"

Physiochemical information

The data quality rule configuration in the group "Physiochemical information" has been removed because it is no longer needed with the changes of the GDSN version. It contained following data quality rule configuration:

- "Check items with 'Physiochemical characteristic value' have a corresponding UOM populated"

Updated data quality rule configurations (IM)

Target market specific checks / Marketing information

"Check 'Coupon family code' has max length of 3 characters (TM: US) and Check 'Coupon family code' has min length of 3 characters (TM: US)

To support the multi value change for the "Coupon family code" field the according data quality rules and data quality rule configurations had to be adjusted.

The data quality rule configurations were based on the data quality rules "Check_MaxLength" and "Check_MinLength". Those have been adjusted to use the new data quality rules "Check_MaxLengthVariableMessage" and "Check_MinLengthVariableMessage". The rule has an additional input port "inMultiValueDelimiter" where the delimiter of the field is defined. See below the change from the previous configuration (left) to the updated configuration (right).

Name*:
Check 'Coupon family code' has max length of 3 character ☐ Hidden

Description:
Checks if the value of the coupon family code is at most 3 characters long for the target market 'US'.

Rule name*:
Check_MaxLength

Data type: Item

Input port	Data type	Field
inLength	string(10)	"3"
inData	string(4096)	Coupon family code (USA)

Data type: Item

Output port	Data type	Field
-------------	-----------	-------

Name*:
Check 'Coupon family code' has max length of 3 character ☐ Hidden

Description:
Checks if the value of the coupon family code is at most 3 characters long for the target market 'US'.

Rule name*:
Check_MaxLengthVariableMessage

Data type: Item


Input port	Data type	Field
inObjectID	string(512)	Object code number
inLength	string(12)	"3"
inCheckField	string(4096)	Coupon family code (USA)
inMultiValueDelimiter	string(10)	","
inStatusMessage	string(50)	

Furthermore the data quality rule configurations got moved from the group "Target market specific checks" to the newly introduced group "Marketing information".

Updated data quality rules

GDSN Data quality rule "Check_MaxLengthVariableMessage"

This rule got an additional input parameter "inMultiValueDelimiter" to be able to support multi value checks. In case the parameter is empty, a single value is expected in the "inCheckField" input port.

 Existing data quality rule configurations must not be adjusted since the single value check is working as before.

New data quality rules

GDSN Data quality rule "Check_MinLengthVariableMessage"

This rule will check if the length of the input string is more than or equal to the defined minimum length. The input string is given in the port "inCheckField" and the length in the port "inLength". In case input string is a multi value field, a delimiter has to be specified to detect single values which are going to be checked. The according port to define the delimiter is called "inMultiValueDelimiter". Furthermore it's possible to override the rules error message with a GDSN specific error message. The default error message is "inCheckField length exceeds the maximum length defined (<inLength>)."

2.1.16.6 Additional changes

Some improvements have been made in addition to the needed changes for GDSN version 3.1.3. This chapter will show up those changes.

New GDSN unit enumeration

There is a new enumeration containing all GDSN units. This enumeration has been assigned to all fields in context of GDSN or food and beverage that used the "all units" enumeration.

Technical information:

- enumeration identifier: "Enum.GDSNUnits"
- identifiers of affected fields:
 - ArticleFishMeatPoultryContentUOM.MinimumFishMeatPoultryContentUOM
 - ArticlePhysioChemicalUOM.PhysiochemicalCharacteristicValueUOM
 - GDSNCanadaExtensionUOM.SuggestedServingSizeUOM
 - ArticleNutrientBasisQuantity.ServingSizeUOM
 - ArticleMicrobiologicsUOM.Unit and corresponding logical key ArticleMicrobiologicsUOM.LK.Unit

New view for Product 360 Desktop client

A new view "Confirmation message" has been introduced. It is used to show the maintained messages for a CIC in the data recipient scenario.

Properties of the view:

- read only
- visible for data recipient scenario only
- dependent on the "Confirmation status" selection

Confirmation status (1)

Article_74824090894562

	Sender	Recipient	Target mar...	Me...	Public...	Linked GTIN	Severity
1	Our GLN	DataProvider - ThaStore	USA	CIC	USA	<no GTIN>	REJEC...
1 element selected							
Sender							

Confirmation messages (1)

Our GLN - DataProvider - ThaStore - USA - CIC - <no GTIN>

	Message
1	You did not maintain the fields as we discussed on the phone. The field "dummy" should be s...
1 element selected	
Message	

New data quality rule configurations

Dairy fish meat poultry information

The data quality rule configuration "Check items with 'Fish meat poultry type code' have a 'Minimum fish meat poultry content' populated" has been added to the group "Dairy Fish Meat Poultry information". It is used to ensure that the field "Minimum fish meat poultry content" is maintained if the "Fish meat poultry type code", "Fish meat poultry type code list agency" or "Fish meat poultry type code list identification" is maintained. Since those fields are all logical keys, the data quality rule configuration is only checking one of them.

See below the new data quality configuration.

Name*: ☐ Hidden

Description:

Rule name*:

Data type:

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Fish meat poultry type code
inCheckField	string(512)	Minimum fish meat poultry content
inStatusMessage	string(50)	

Data type:

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

Adjusted data quality rule configurations

Nutritional information

The existing data quality rule configuration "Check 'Daily value intake reference' or 'Quantity contained' is not empty if 'Nutrient type' is set" configuration has been adjusted according to the GDSN validation.

- Previous configuration name: Check 'Daily value intake reference' or 'Quantity contained' is not empty if 'Nutrient type' is set
- New configuration name: Check 'Percentage of daily intake' or 'Quantity contained' is not empty if 'Nutrient type' is set
- Previous input port mapping for inCheckField1: Daily value intake reference
- New input port mapping for inCheckField1: Percentage of daily intake

Please find below the changes from the previous versions on the left side and the new configuration on the right side.

Name*:
Check 'Daily value intake reference' or 'Quantity contained' is not ☐ Hidden

Description:
Checks if at least one of the two fields 'Daily value intake reference' or 'Quantity contained' are not empty if 'Nutrient type' is set.

Rule name*:
Check_ConditionalNotEmptyAtLeastOneNotEmpty

Data type: Nutritional information

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Nutrient type
inCheckField1	string(512)	<u>Daily value intake reference</u>
inCheckField2	string(512)	Quantity contained
inCheckField3	string(512)	
inCheckField4	string(512)	
inStatusMessage	string(50)	"GDSN1043"

Data type: Item

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

Name*:
Check 'Percentage of daily intake' or 'Quantity contained' is no ☐ Hidden

Description:
Checks if at least one of the two fields 'Percentage of daily intake' or 'Quantity contained' are not empty if 'Nutrient type' is set.

Rule name*:
Check_ConditionalNotEmptyAtLeastOneNotEmpty

Data type: Nutritional information

Input port	Data type	Field
inObjectID	string(512)	Object code number
inConditionField	string(512)	Nutrient type
inCheckField1	string(512)	<u>Percentage of daily intake</u>
inCheckField2	string(512)	Quantity contained
inCheckField3	string(512)	
inCheckField4	string(512)	
inStatusMessage	string(50)	"GDSN1043"

Data type: Item

Output port	Data type	Field
outObjectID	string(512)	Object code number
outStatusMessage	string(4096)	Message
outStatusCode	string(10)	Status

2.1.16.7 Valid value list changes

AllergenTypeCode				
Used in field ArticleAllergenList.AllergenTypeCode (Allergen type, Allergenart), context: food & beverage				
Code	Name (en)	Name (de)	Action	Remarks
TN	Contains traces of tree nuts (TN)	Enthält Spuren von Nüssen (TN)	removed	manual check and migration needed
GrapeVarietyCode				
Used in field Ingredient.GrapeVarietyCode (Grape variety code, Rebsortencode), context: food & beverage				
Code	Name (en)	Name (de)	Action	Remarks
ARNEIS	Arneis	Arneis	added	DSE only
BLACK QUEEN	Black Queen	Black Queen	added	DSE only
DELAWARE	Delaware	Delaware	added	DSE only
KOSHU	Koshu	Koshu	added	DSE only
MUSCAT BAILEY A	Muscat Bailey A	Muscat Bailey A	added	DSE only
NIAGARA	Niagara	Niagara	added	DSE only
REGENT	Regent	Regent	added	DSE only
RYUGAN	Ryugan	Ryugan	added	DSE only
SMÊDEREVKA	Smederevka	Smederevka	added	DSE only

TERRET	Terret	Terret	add ed	DSE only
VRANEC	Vranec	Vranec	add ed	DSE only
NutrientTypeCode Used in field ArticleNutrientList.NutrientTypeCode (Nutrient type, Nährstoffart), context: food & beverage				
Code	Name (en)	Name (de)	Action	Remarks
X_FATRSA	Combined saturated fat and trans fat (X_FATRSA)	Gesättigte und ungesättigte Fette kombiniert (X_FATRSA)	add ed	
X_CASNWH	Casein to whey protein ratio (X_CASNWH)	Molkeneiweiß-Casein-Verhältnis (X_CASNWH)	add ed	DSE only
X_FUNS	Total unsaturated fat (X_FUNS)	Ungesättigte Fettsäuren insgesamt (X_FUNS)	add ed	DSE only
X_PROSOL	Soluble protein (X_PROSOL)	Lösliches Eiweiß (X_PROSOL)	add ed	DSE only
X_SALATRIM	Salatrim (X_SALATRIM)	Salatrim (X_SALATRIM)	add ed	DSE only
X_SUCRA	Sucralose (X_SUCRA)	Sucralose (X_SUCRA)	add ed	DSE only
NutritionalClaimNutrientElementCode Used in field ArticleNutritionalClaim.NutritionalClaimNutrientElementCode (Nutritional claim element code, Nährwertanspruch: Nährstoffcode), context: food & beverage				
Code	Name (en)	Name (de)	Action	Remarks
CAFFEINE	Caffeine	Koffein	add ed	
DECAFFEINATED	Decaffeinated	Entkoffeiniert	add ed	
ARTIFICIAL_COLOUR	Artificial colour	Künstliche Farbstoffe	add ed	IM only

ASPARTAME	Aspartame	Aspartam	add ed	IM only
AZO_DYE	Azo dye	Azofarbstoff	add ed	IM only
BEEF_GELATINE	Beef gelatine	Rindergelatine	add ed	IM only
FLAVOUR_ENHANCERS	Flavour enhancers	Geschmacksverstärker	add ed	IM only
GELATINE	Gelatine	Gelatine	add ed	IM only
HIGH_FRUCTOSE_CORN_SYRUP	High fructose corn syrup	Maissirup mit hohem Fruchtzuckergehalt	add ed	IM only
MSG	Monosodium glutamate	Mononatriumglutamat	add ed	IM only
NANOPARTICLE	Nanoparticle	Nanopartikel	add ed	IM only
NATURAL_FLAVOUR	Natural flavour	Natürliches Aroma	add ed	IM only
NITRATE	Nitrate	Nitrat	add ed	IM only
PHENYLALANINE	Phenylalanine	Phenylalanin	add ed	IM only
PHOSPHATE	Phosphate	Phosphat	add ed	IM only
PORK_GELATINE	Pork gelatine	Schweinegelatine	add ed	IM only
SMOKE_FLAVOUR	Smoke flavour	Raucharoma	add ed	IM only
SODIUM_NITRITE	Sodium nitrite	Natriumnitrit	add ed	IM only
SORBITOL	Sorbitol	Sorbitol	add ed	IM only
WATER	Water	Wasser	add ed	IM only

NutritionalClaimTypeCode

Used in field ArticleNutritionalClaim.NutritionalClaimTypeCode (Nutritional claim type code, Nährwertanspruch: Code der Art), context: food & beverage

Code	Name (en)	Name (de)	Action	Remarks
PURE	Pure	Pur	added	
REAL	Real	Echt	added	

OrganicClaimAgency

Used in field IngredientOrganic.ClaimAgencyCode (Organic claim agency code, Ökologische Kontrollstelle), context: food & beverage

Code	Name (en)	Name (de)	Action	Remarks
7	Quality Assurance International (Deprecated)	Quality Assurance International (Deprecated)	changed	set to deprecated
8	Südtirol Bioland (Deprecated)	Südtirol Bioland (Deprecated)	changed	set to deprecated
9	Ecoland (Deprecated)	Ecoland (Deprecated)	changed	set to deprecated
12	Naturland (Deprecated)	Naturland (Deprecated)	changed	set to deprecated
13	BVÖL (Deprecated)	BVÖL (Deprecated)	changed	set to deprecated
14	ECOVIN (Deprecated)	ECOVIN (Deprecated)	changed	set to deprecated
15	IFOAM (Deprecated)	IFOAM (Deprecated)	changed	set to deprecated

19	QCS (Deprecated)	QCS (Deprecated)	changed	set to deprecated
PreparationType Used in field ArticlePreparationServing.PreparationType (Preparation type, Zubereitungsart), context: food & beverage				
Code	Name (en)	Name (de)	Action	Remarks
AIR_FRY	Air frying	Heißluftfritieren	added	IM only
OrganismCode Used in field ArticleMicrobiologics.OrganismCode (Organism code, Organismus-Code), context: food & beverage				
Code	Name (en)	Name (de)	Action	Remarks
MILK_ACID	Milk acid	Milchsäurebakterien	changed	changed code from MILK-ACID to MILK_ACID
BACILLUS_SUBTILIS	Bacillus subtilis (hay or grass bacillus)	Bacillus subtilis (Heu- oder Grasbazillus)	added	
HISTAMINE	Histamine	Histamine	added	
NOROVIRUS	Norovirus	Norovirus	added	
NutritionalProgramCode Used in field ArticleHealthCare.NutritionalProgramCode (Nutritional program code, Code des Ernährungsprogramms), context: food & beverage				
Code	Name (en)	Name (de)	Action	Remarks
2	USDA MyPlate	USDA MyPlate	added	
3	Dietary Guidelines for Americans	Ernährungsrichtlinien für Amerikaner	added	

4	Fruits & Veggies - More Matters	Obst & Gemüse - Es zählt mehr	added	
5	3-Every-Day of Dairy	3 Milchprodukte jeden Tag	added	
6	American Diabetes Association Food Exchange List	Essensliste der Amerikanischen Diabetes-Vereinigung	added	
7	Weight Watchers	Weight Watchers	added	

SurfaceOfCheeseAtEndOfRipeningCode

Used in field ArticleCheese.SurfaceOfCheeseAtEndOfRipeningCode (Surface of cheese at end of ripening, Käseoberfläche am Ende der Reifezeit), context: food & beverage

Code	Name (en)	Name (de)	Action	Remarks
SOFT_RIPENED_MOULD_RIND	Soft-ripened mould rind	Weich-gereifte Schimmelrinde	added	

PhysiochemicalCharacteristicCode

Used in field ArticlePhysioChemical.PhysiochemicalCharacteristicCode (Physiochemical characteristic code, Physiochemische Eigenschaft), context: food & beverage

Code	Name (en)	Name (de)	Action	Remarks
AFLATOXIN_B1_LESS_THAN	Aflatoxin B1	Aflatoxin B1	changed	changed code from AFLATOXIN_B1_LESS_THAN to AFLATOXIN_B1_LESS_THAN
BASE_QUANTITY_PER_PACKAGE	Base quantity per package	Basismenge pro Packung	added	
PURE_BASE_RATIO	Pure base ratio	Reinanteil	added	

ApplicationIdentifierType

Used in field ArticleDataCarrier.ApplicationIdentifierType (Application identifier type, Application identifier), context: GDSN

Code	Name (en)	Name (de)	Action	Remarks
21	Serial number	Seriennummer	added	IM only
PackagingTypeCode Used in field ArticlePackaging.PackagingType (Packaging type, Verpackungsart), context: GDSN				
Code	Name (en)	Name (de)	Action	Remarks
CQ	Cartridge	Kartusche	added	
RO	Roll	Rolle	added	
PackagingMaterialTypeCode Used in field ArticlePackagingMaterial.PackagingMaterialTypeCode (Packaging material type code, Verpackungsmaterialtyp), context: GDSN				
Code	Name (en)	Name (de)	Action	Remarks
METAL_COMPOSITE	Metal composite	Metallverbund	added	IM only
SustainabilityFeatureCode Used in field ArticleSustainability.TradeItemSustainabilityFeatureCode (Sustainability feature code, Nachhaltigkeitseigenschaft), context: GDSN				
Code	Name (en)	Name (de)	Action	Remarks
MADE_FROM_RECYCLED_MATERIAL	Recycled material	Recyclingmaterial	added	
REUSABLE	Reusable	Wiederverwendbar	added	
AdditionalPartyIDList Used in field ThirdPartyAdditionalIdentification.Type (Third party is used in Certification.Organisation), context: general				
Code	Name (en)	Name (de)	Action	Remarks

EU_VAT_IDENTIFICATION_NUMBER	EU VAT Identification Number	EU Umsatzsteuer-Identifikationsnummer	added	
ReferencedFileTypeCode Used in field CertificateReferencedFile.FileTypeCode(Referenced file type, Typ der referenzierten Datei), context: general				
Code	Name (en)	Name (de)	Action	Remarks
360_DEGREE_IMAGE	360 Degree Image	360 Grad Bild	added	
ASSEMBLY_INSTRUCTIONS	Assembly Instructions	Aufbau-/Montageanleitung	added	
PACKAGING_ARTWORK	Packaging Artwork	Verpackungs-Bebilderung	added	
QR_CODE	Link to QR Code	Verknüpfung zum QR URL/URI	added	
Units Used in several UOM fields, context: GDSN units in general				
Code	Name (en)	Name (de)	Action	Remarks
RO	roll	Rolle	changed	changed code from RL to RO
XRO	Roll (GDSN)	Rolle (GDSN)	added	category: Measuring units
MTS	Metre per second	Meter pro Sekunde	added	category: Measuring units
AM	Ampoule (deprecated)	Ampulle (veraltet)	changed	removed category from this unit, so it won't show up in any unit enumeration which is based on this category; changed name to mark the unit as

				deprecated
MPN	most probable number	Wahrscheinlichste Anzahl	changed	added category: Microbiological units, available in "Enum.GDSNMicrobiologicalUnits" enumeration
CFU	colony forming units	Koloniebildende Einheiten	changed	added category: Microbiological units, available in "Enum.GDSNMicrobiologicalUnits" enumeration

2.1.16.8 DSE Export template changes

This chapter lists all changes that have been made to the export templates used for DSE scenarios.

General changes

Renaming of export templates

The export templates got more convenient names. The names of the export files have been adjusted as well, they are the same as those of the export templates.

Data source

- MjR3_DSE_CatalogItemConfirmation → CIC_CatalogItemConfirmation
- MjR3_DSE_CatalogItemSubscription → CIS_CatalogItemSubscription
- MjR3_DSE_RequestForCatalogItem → RFCIN_RequestForCatalogItem

Data recipient

- MjR3_DSE_CatalogItemNotification → CIN_CatalogItemNotification
- MjR3_DSE_CatalogItemPublication → CIP_CatalogItemPublication
- MjR3_DSE_CatalogItemPublicationWithdrawal → CIPHW_CatalogItemPublicationHierarchyWithdrawal

General export template changes

The following changes have been made to all export templates:

- Adjustments according to new XSD files
- Post export step "Validate XML file(s)": cancel export in case of error during XML validation
- The post export step "XML Pretty print" has been removed. It should only be used during the implementation and testing phase but not in production environment.

New XSD file set

A new XSD file set has been provided for GDSN version 3.1.3. All old XSD files used for export post steps have to be deleted, then all new XSD files must be uploaded.

Data recipient export templates

All data recipient export templates (Catalog item confirmation, Catalog item subscription, Request for catalog item) have to be replaced with the new export templates.

Data source export templates

The item publication export templates (Catalog item publication, Catalog item publication hierarchy withdrawal) must be replaced by the corresponding new export templates.

If you use the provided default "Catalog item notification" export template you should replace it by the new one as well. In case this is not possible because of customizations, you'll find a detailed description of changes below.

Catalog item notification export template

The "Catalog item notification" export template is the most complex export template of the GDSN accelerator package. The following section describes the changes that have been made to that export template in detail and thus enables you to apply those changes to any customized "Catalog item notification" export template.

File header

The "documentCommandType" variable has been renamed from "Operation (ADD, CHANGE_BY_REFRESH, CORRECT or DELETE)" to "Operation (ADD, CHANGE_BY_REFRESH or CORRECT)". The option "DELETE" was removed from the variable name because it's not valid anymore.

TradeItemDescriptionModule

Affected field: "GDSN short description"

The existing export validation for truncating the value to a maximum length of 35 for the export fields "Item.GDSN short description (Target market, Language)" and "Item.GDSN short description (Target market, Language second)" has been adjusted in the export format template. From now on a warning instead of an error will be logged in case the truncate of the value is executed.

MarketingInformationModule

Affected fields: "Marketing message", "SpecialItemCode", "CouponFamilyCode"

The field "Marketing message" has been moved from "Marketing information" to new export sub-module "Marketing messages". In addition, multiple values for "CouponFamilyCode" can be exported from now on.

Previous DSE export template snippet, call of "Marketing information" sub-module ("Items" main module)

```

1  {{CreateXMLTagWithContent {$Marketing information}
2    , "<marketing_information:marketingInformationModule xsi:schemaLocation="
    urn:gs1:gdsn:marketing_information:xsd:3 http://www.gdsregistry.org/3.1/
    schemas/gsl/gdsn/MarketingInformationModule.xsd" xmlns:xsi="http://
    www.w3.org/2001/XMLSchema-instance" xmlns:marketing_information="urn:gs1:g
    dsn:marketing_information:xsd:3">
3      <marketingInformation>
4        ?
5      </marketingInformation>
6    </marketing_information:marketingInformationModule>"
7  }
```

Previous DSE export template content for sub-module "Marketing information"

```

1  {{CreateXMLTagWithValue {{FormatDecimal {{Item.Coupon family code (Target
2    market)}},"",0}
3    , "<couponFamilyCode>?</couponFamilyCode>"}
4  {{SplitKeywords {{EnumerationKey {{Item.Special item code (Target
5    market)}}
6    , <specialItemCode>, </specialItemCode>}}
7    {{!No second Language allowed}
8    {{CreateXMLTagWithValue {{Item.Marketing message (Target market,
9    Language)}}
    ,<tradeItemMarketingMessage languageCode="{{GDSNEnumerationCode
    {{Item.Language (Target market, Language)},"GDSN"}}">?</
    tradeItemMarketingMessage>}}
```

New DSE export template snippet, call of "Marketing information" and "Marketing messages" sub-modules ("Items" main module)

```

1  {{CreateXMLTagWithContent {{Concat {$Marketing information}, {$Marketing
2    messages}}}
3    , "<marketing_information:marketingInformationModule xsi:schemaLocation="
    urn:gs1:gdsn:marketing_information:xsd:3 http://www.gdsregistry.org/3.1/
    schemas/gsl/gdsn/MarketingInformationModule.xsd" xmlns:xsi="http://
    www.w3.org/2001/XMLSchema-instance" xmlns:marketing_information="urn:gs1:g
    dsn:marketing_information:xsd:3">
4      <marketingInformation>
5        ?
6      </marketingInformation>
    </marketing_information:marketingInformationModule>"
```

7	}
---	---

New DSE export template content for sub-module "Marketing information"

1	{?SplitKeywords {?FormatDecimal {&Item.Coupon family code (Target market)}}, "", 0}
2	, <couponFamilyCode>, </couponFamilyCode>}
3	
4	{?SplitKeywords {?EnumerationKey {&Item.Special item code (Target market)}}}
5	, <specialItemCode>, </specialItemCode>}

New DSE export template content for sub-module "Marketing messages"

1	{?CreateXMLTagWithValue {&Marketing messages.Marketing message},
2	"<tradeItemMarketingMessage sequenceNumber=\"{?FormatDecimal {&Marketing messages.Sequence}}, \"\", 0}\"
3	languageCode=\"{?GDSNEnumerationCode {&Marketing messages.Language}}, \"GDSN\"}\">?</tradeItemMarketingMessage>"}}

DairyFishMeatPoultryItemModule

Affected field: "Cheese maturation period description"

As multi values are supported now, a corresponding export field has been added to output the second language value as well.

Previous DSE export template snippet for "Dairy fish meat poultry" sub-module ("Items" main module)

1	, {?Concat "<cheeseInformation>"
2	{!No second Language allowed}
3	, {?CreateXMLTagWithValue {&Dairy fish meat poultry.Cheese maturation period description (Language)}
4	, "<cheeseMaturationPeriodDescription languageCode=\"{?GDSNEnumerationCode {&Dairy fish meat poultry.Language (Language)}}, \"GDSN\"}\">?</cheeseMaturationPeriodDescription>"}}

New DSE export template snippet for "Dairy fish meat poultry" sub-module ("Items" main module)

1	, {?Concat "<cheeseInformation>"
2	, {?CreateXMLTagWithValue {&Dairy fish meat poultry.Cheese maturation period description (Language)}

```

3      , "<cheeseMaturationPeriodDescription languageCode=\"{"?
GDSNEnumerationCode {&Dairy fish meat poultry.Language (Language)},
"GDSN"}\">?</cheeseMaturationPeriodDescription>"}
4      , {?IfNotEmptyThen {?ValueGet "LanguageDiffs"}, {?CreateXMLTagWithValue
{&Dairy fish meat poultry.Cheese maturation period description (Language
second)}}
5      , "<cheeseMaturationPeriodDescription languageCode=\"{"?
GDSNEnumerationCode {&Dairy fish meat poultry.Language (Language second)},
"GDSN"}\">?</cheeseMaturationPeriodDescription>"}"}

```

FoodAndBeveragePropertiesInformationModule

Microbiological information

Affected fields: "Organism maximum value", "Organism reference value", "Organism warning value", "Unit"

Limitation: If you maintain entries of the same unit (this is possible for different UOM types) for microbiological details information, then there will be no check in the export which is preventing that such data will be exported. The transfer of duplicate entries would cause an error in the GDSN pool.

Previous DSE export template snippet, call of "Microbiological information" sub-module ("Items" main module)

```

1  {?CreateXMLTagWithContent {?Concat {$Microbiological information},
2  {$Physiochemical information}}
3  , "<food_and_beverage_properties_information:foodAndBeveragePropertiesInf
ormationModule xsi:schemaLocation="urn:gs1:gdsn:food_and_beverage_properti
es_information:xsd:3 http://www.gdsregistry.org/3.1/schemas/gsl/gdsn/
FoodAndBeveragePropertiesInformationModule.xsd" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance"
xmlns:food_and_beverage_properties_information=
"urn:gs1:gdsn:food_and_beverage_properties_information:xsd:3">
4  ?
5  </food_and_beverage_properties_information:foodAndBeveragePropertiesIn
formationModule>"
}

```

Previous DSE export template content for sub-module "Microbiological information"

```

1  <microbiologicalInformation>
2  <microbiologicalOrganismCode>{?EnumerationKeyStandard {&Microbiological
information.Organism code}}</microbiologicalOrganismCode>
3  {!No second UOM allowed}
4  {?CreateXMLTagWithValue {?FormatDecimal {&Microbiological
information.Organism maximum value (UOM type)}}, ".", 6}
5  , "<microbiologicalOrganismMaximumValue
measurementUnitCode=\"{"&Microbiological information.Organism maximum
value UOM (UOM type).Code (Unit system)}\">?</
microbiologicalOrganismMaximumValue>"}

```

```

6      {!No second UOM allowed}
7      {?CreateXMLTagWithValue {?FormatDecimal {&Microbiological
information.Organism reference value (UOM type)}}, ".", 6}
8      , "<microbiologicalOrganismReferenceValue
measurementUnitCode=\"{&Microbiological information.Organism reference
value UOM (UOM type).Code (Unit system)}\">?</
microbiologicalOrganismReferenceValue>"}
9      {!No second UOM allowed}
10     {?CreateXMLTagWithValue {?FormatDecimal {&Microbiological
information.Organism warning value (UOM type)}}, ".", 6}
11     , "<microbiologicalOrganismWarningValue
measurementUnitCode=\"{&Microbiological information.Organism warning
value UOM (UOM type).Code (Unit system)}\">?</
microbiologicalOrganismWarningValue>"}
12    </microbiologicalInformation>

```

New DSE export template snippet, call of "Microbiological information" and "Microbiological information: Collect details" sub-modules ("Items" main module)

```

1      {$Microbiological information: Collect details}
2      {?CreateXMLTagWithContent {?Concat {$Microbiological information},
{$Physiochemical information}}
3      , "<food_and_beverage_properties_information:foodAndBeveragePropertiesInf
ormationModule xsi:schemaLocation=\"urn:gs1:gdsn:food_and_beverage_properti
es_information:xsd:3 http://www.gdsregistry.org/3.1/schemas/gsl/gdsn/
FoodAndBeveragePropertiesInformationModule.xsd\" xmlns:xsi=\"http://
www.w3.org/2001/XMLSchema-instance\"
xmlns:food_and_beverage_properties_information=
\"urn:gs1:gdsn:food_and_beverage_properties_information:xsd:3\">
4      ?
5      </food_and_beverage_properties_information:foodAndBeveragePropertiesIn
formationModule>\"
6      }

```

New DSE export template content for sub-module "Microbiological information"

```

1      {?ValueSet "tmpOrganismCode", {?EnumerationKeyStandard {&Microbiological
information.Organism code}}}
2      <microbiologicalInformation>
3      <microbiologicalOrganismCode>{?ValueGet "tmpOrganismCode"}</
microbiologicalOrganismCode>
4      {?IfEmptyThenNotEnc "", {?ValueGet {?ValueGet "tmpOrganismCode"}
_maxValue}}
5      {?IfEmptyThenNotEnc "", {?ValueGet {?ValueGet "tmpOrganismCode"}
_refValue}}
6      {?IfEmptyThenNotEnc "", {?ValueGet {?ValueGet "tmpOrganismCode"}
_warnValue}}
7      </microbiologicalInformation>

```

```

8
9      {!Reset variables}
10     {?ValueSet "tmpOrganismCode", ""}
11     {?ValueSet {?ValueGet "tmpOrganismCode"}_maxValue, ""}
12     {?ValueSet {?ValueGet "tmpOrganismCode"}_refValue, ""}
13     {?ValueSet {?ValueGet "tmpOrganismCode"}_warnValue, ""}

```

New DSE export template content for sub-module "Microbiological information: Collect details"

```

1      {?ValueSet "tmpOrganismCode", {?EnumerationKeyStandard {&Microbiological
2      detail information.Organism code}}}}
3      {!Organism maximum value}
4      {?ValueSet
5      {!name of variable to set}{?ValueGet "tmpOrganismCode"}_maxValue,
6      {!existing value}{?ValueGet {?ValueGet "tmpOrganismCode"}_maxValue}
7      {!add new value}{?CreateXMLTagWithValue {?FormatDecimal
8      {&Microbiological detail information.Organism maximum value},.,6},
9      "<microbiologicalOrganismMaximumValue
10     measurementUnitCode=\"{&Microbiological detail information.Unit.Code
11     (Unit system)}\">?</microbiologicalOrganismMaximumValue>"}
12     }
13     {!Organism reference value}
14     {?ValueSet
15     {!name of variable to set}{?ValueGet "tmpOrganismCode"}_refValue,
16     {!existing value}{?ValueGet {?ValueGet "tmpOrganismCode"}_refValue}
17     {!add new value}{?CreateXMLTagWithValue {?FormatDecimal
18     {&Microbiological detail information.Organism reference value},.,6},
19     "<microbiologicalOrganismReferenceValue
20     measurementUnitCode=\"{&Microbiological detail information.Unit.Code
21     (Unit system)}\">?</microbiologicalOrganismReferenceValue>"}
22     }
23     {!Organism warning value}
24     {?ValueSet
25     {!name of variable to set}{?ValueGet "tmpOrganismCode"}_warnValue,
26     {!existing value}{?ValueGet {?ValueGet "tmpOrganismCode"}_warnValue}
27     {!add new value}{?CreateXMLTagWithValue {?FormatDecimal
28     {&Microbiological detail information.Organism warning value},.,6},
29     "<microbiologicalOrganismWarningValue
30     measurementUnitCode=\"{&Microbiological detail information.Unit.Code
31     (Unit system)}\">?</microbiologicalOrganismWarningValue>"}
32     }

```

Physiochemical information

Affected fields: "Physiochemical characteristic value UOM"

The field is no longer mandatory; if the UOM is not maintained the attribute measurementUnitCode="<unit>" won't be added to the XML tag.

Previous DSE export template content for "Physiochemical information" sub-module

```

1  <physiochemicalCharacteristic>
2    <physiochemicalCharacteristicCode>{?EnumerationKey {&Physiochemical
information.Physiochemical characteristic code}}</
physiochemicalCharacteristicCode>
3    {?CreateXMLTagWithValue {?FormatDecimal {&Physiochemical
information.Physiochemical characteristic value (UOM type)}}, ".", 6}
4    , "<physiochemicalCharacteristicValue
measurementUnitCode=\"{&Physiochemical information.Physiochemical
characteristic value UOM (UOM type).Code (Unit system)}\">?</
physiochemicalCharacteristicValue>"}
5    {?IfNotEmptyThenNotEnc {?ValueGet "UOMTypeDiffs"}
6    , {?Compare {&Physiochemical information.Physiochemical characteristic
value UOM (UOM type second).Code (Unit system)}},{&Physiochemical
information.Physiochemical characteristic value UOM (UOM type).Code (Unit
system)}},,
7    {?CreateXMLTagWithValue {?FormatDecimal {&Physiochemical
information.Physiochemical characteristic value (UOM type second)}}, ".",
6}
8    , "<physiochemicalCharacteristicValue
measurementUnitCode=\"{&Physiochemical information.Physiochemical
characteristic value UOM (UOM type second).Code (Unit system)}\">?</
physiochemicalCharacteristicValue>"}
9    }
10   }
11  </physiochemicalCharacteristic>

```

New DSE export template content for "Physiochemical information" sub-module

```

1  {?ValueSet "outputCode", {?EnumerationKeyStandard {&Physiochemical
information.Physiochemical characteristic code}}}
2  {?ValueSet "outputValueFirst", {?FormatDecimal {&Physiochemical
information.Physiochemical characteristic value (UOM type)}}, ".", 6}}
3  {?ValueSet "outputValueSecond", {?FormatDecimal {&Physiochemical
information.Physiochemical characteristic value (UOM type second)}}, ".",
6}}
4  {?ValueSet "uomFirst", {&Physiochemical information.Physiochemical
characteristic value UOM (UOM type).Code (Unit system)}}
5  {?ValueSet "uomSecond", {&Physiochemical information.Physiochemical
characteristic value UOM (UOM type second).Code (Unit system)}}
6
7  {?IfNotEmptyThenNotEnc {?ValueGet "outputCode"}{?ValueGet
"outputValueFirst"}{?ValueGet "outputValueSecond"},
8  "<physiochemicalCharacteristic>
9  {?CreateXMLTagWithValue {?ValueGet "outputCode"},
"<physiochemicalCharacteristicCode>?</physiochemicalCharacteristicCode>"}
10
11  {! value and unit for uom type}

```



```

12  {?ValueSet "uomQual", {?CreateXMLTagWithValue {?ValueGet "uomFirst"},
13  "measurementUnitCode=\"?\""}}
14  {?CreateXMLTagWithValue {?ValueGet "outputValueFirst"}, {?Concat
15  "<physiochemicalCharacteristicValue ", {?ValueGet "uomQual"}, ">?</
16  physiochemicalCharacteristicValue>"}
17  }
18  {! value and unit for uom type second, only if not equal to uom type and
19  uom differs}
20  {?IfNotEmptyThenNotEnc {?ValueGet "UOMTypeDiffs"}, {?Compare {?ValueGet
21  "uomFirst"}, {?ValueGet "uomSecond"}, "",
22  {?ValueSet "uomQual", {?CreateXMLTagWithValue {?ValueGet "uomSecond"},
23  "measurementUnitCode=\"?\""}}
24  {?CreateXMLTagWithValue {?ValueGet "outputValueSecond"}, {?Concat
25  "<physiochemicalCharacteristicValue ", {?ValueGet "uomQual"}, ">?</
26  physiochemicalCharacteristicValue>"}
27  }
28  }}
29  </physiochemicalCharacteristic>"
30  }

```

2.1.16.9 IM Export template changes

This chapter lists all changes that have been made to the export templates used for IM scenarios.

General changes

Renaming of export templates

The export templates got more convenient names. The names of the export files have been adjusted as well, they are the same as those of the export templates.

Data source

- 1Sync CatalogItemConfirmation → CIC_CatalogItemConfirmation
- 1Sync CatalogItemSubscription → CIS_CatalogItemSubscription

Data recipient

- MjR3_1Sync CatalogueRequest Item → CR_CatalogueRequest Item
- 1Sync CatalogueRequest Link ADD → CR_CatalogueRequest Link ADD
- 1Sync CatalogueRequest Link DELETE → CR_CatalogueRequest Link DELETE
- 1Sync CatalogueRequest Publication → CR_CatalogueRequest Publication
- 1Sync CatalogueRequest Publication_PublicationWithdrawal → CR_CatalogueRequest Publication HW
- 1Sync CatalogueRequest Publication_WithoutStatus → CR_CatalogueRequest Publication_ByVariables

General export template changes

The following changes have been made to all export templates:

- Adjustments according to new XSD files
- Changed the envelope parameters for namespace reference and schema location in the module "Header" of each provided export format template

- Post export step "Validate XML file(s)": cancel export in case of error during XML validation
- The post export step "XML Pretty print" has been removed. It should only be used during the implementation and testing phase but not in production environment.

New XSD file set

A new XSD file set has been provided for GDSN version 3.1.3. All old XSD files used for export post steps have to be deleted, then all new XSD files must be uploaded.

Data recipient export templates

All data recipient export templates (Catalog item confirmation, Catalog item subscription) have to be replaced with the new export templates.

Data source export templates

The item publication export templates (Catalog request publication, Catalog request publication hierarchy withdrawal) and the item link export templates (Link ADD, Link DELETE) must be replaced by the corresponding new export templates.

If you use the provided default "Catalog request item" export template you should replace it by the new one as well. In case this is not possible because of customizations, you'll find a detailed description of changes below.

Catalog request item template

The "Catalog request item" export template is the most complex export template of the GDSN accelerator package. The following section describes the changes that have been made to that export template in detail and thus enables you to apply those changes to any customized "Catalog request item" export template.

General

Some export fields got renamed, e.g.: from {&Dairy fish meat poultry.Language(1)} to {&Dairy fish meat poultry.Language (Language)} to improve readability. Some export sub-modules have been moved up or down within the export template to bring them into a more logical order.

Note: Neither the names of export data fields nor the order of export sub-modules have any impact on the output file.

TargetMarketAttributes

Affected field: "GDSN short description"

A data field validation for the export fields "Item.GDSN short description (Target market, Language)" and "Item.GDSN short description (Target market, Language second)" has been added to the export template. All values will be truncated to a maximum length of 35 characters, and a warning will be logged to the export protocol.

Flex attributes: couponFamilyCode, specialItemCode

Affected fields: "Special item code", "Coupon family code"

The output of those fields has been moved to new sub-module "Marketing Information". In addition, multiple values will be exported for "Coupon family code" now.

Previous IM export template, call of "Coupon family code" ("Items" main export module)

```
1  {{CreateXMLTagWithValue {&Item.Coupon family code (Target market)}},
   "<couponFamilyCode>?</couponFamilyCode>"}
```

Previous IM export template, call of "Special item code" ("Items" main export module)

```
1  {{CreateXMLTagWithValue {&Item.Coupon family code (Target market)}},
   "<couponFamilyCode>?</couponFamilyCode>"}
```

New IM export template, call of "Marketing Information" sub-module ("Items" main export module, line 147)

```
1  {{Collect marketing messages}
2  {!start flex}
3  {{CreateXMLTagWithContent
4    {{Concat
5      {{Marketing Information},
6      {{CreateXMLTagWithContent {{Marketing messages}}, "<attrGroupMany name='ma
marketingMessage'>?</attrGroupMany>"}}
7    {{CreateXMLTagWithContent
8      {{SplitKeywords {{GDSNEEnumerationCode {&Item.Countries of origin
(Target market)}}, "1WS"}}
9      , "<row><attr name='countryCode'>", "</attr></row>"}}
10     , "<attrGroupMany name='countryOfOrigin'>?</attrGroupMany>"}}
```

New IM export template, content of "Marketing Information" sub-module

```
1  {{CreateXMLTagWithContent
2    {{SplitKeywords {&Item.Coupon family code (Target market)}}, "<value>",
   "</value>"},
3    "<attrMany name='couponFamilyCode'>?</attrMany>"}}
4
5  {{CreateXMLTagWithContent
6    {{SplitKeywords {{EnumerationKey {&Item.Special item code (Target
market)}}}, "<value>", "</value>" }},
```

7

```
"<attrMany name="specialItemCode">?</attrMany>" }
```

Flex attributes: marketingMessage

Affected field: "Marketing message"

Field "Marketing message" has been moved from "Items" main export module to new export sub-modules "Marketing messages" and "Collect marketing messages".

Previous IM export template, call of "Marketing message" ("Items" main export module)

```
1  {?CreateXMLTagWithValue {&Item.Marketing message (Target market,
2  Language)},
3  "<tradeItemMarketingMessage lang=\"{?GDSNEnumerationCode {&Item.Language
4  (Target market, Language)}, \"GDSN\"}\">?</tradeItemMarketingMessage>"}
5  {?IfNotEmptyThenNotEnc {?ValueGet "LanguageDiffers"},
6  {?CreateXMLTagWithValue {&Item.Marketing message (Target market,
7  Language second)},
8  "<tradeItemMarketingMessage lang=\"{?GDSNEnumerationCode {&Item.Language
9  (Target market, Language second)}, \"GDSN\"}\">?</tradeItemMarketingMessage>
10 "}"
11 }
```

New IM export template, call of "Marketing messages" and "Collect marketing messages" sub-modules ("Items" main export module, lines 143 and 148)

```
1  {$Collect marketing messages}
2  {!start flex}
3  {?CreateXMLTagWithContent
4  {?Concat
5  {$Marketing Information},
6  {?CreateXMLTagWithContent {$Marketing messages},"<attrGroupMany name="ma
7  rketingMessage">?</attrGroupMany>"}
8  {?CreateXMLTagWithContent
9  {?SplitKeywords {?GDSNEnumerationCode {&Item.Countries of origin
10 (Target market)}, "1WS"}
11 , "<row><attr name="countryCode">","</attr></row>"}
12 , "<attrGroupMany name="countryOfOrigin">?</attrGroupMany>"}
13 }
```

New IM export template, content of "Marketing messages" sub-module

```
1  {?IfNotEmptyThenNotEnc {?ValueGet "MarketingMessage_"{&Marketing
2  messages.Sequence}}},
3  <row>
4  <attrQualMany name="tradeItemMarketingMessage">
```

```

4      {?IfEmptyThenNotEnc "", {?ValueGet "MarketingMessage_"&Marketing
5      messages.Sequence}}}}
6      </attrQualMany>
7      {?CreateXMLTagWithValue {?FormatDecimal &Marketing
8      messages.Sequence}, "", 0}, "<attr name='sequenceNumber'>?</attr>"}
9      {!Reset variable}{?ValueSet "MarketingMessage_"&Marketing
      messages.Sequence}, ""}
      </row>
    }

```

Flex attributes: foodAndBevMicrobiological

Affected fields: "Organism maximum value", "Organism reference value", "Organism warning value", "Unit"

Limitation: If you maintain entries of the same unit (this is possible for different UOM types) for microbiological details information, then there will be no check in the export which is preventing that such data will be exported. The transfer of duplicate entries would cause an error in the GDSN pool.

Previous IM export template, content of "Microbiological information" sub-module

```

1  <row>
2  {?CreateXMLTagWithValue {?EnumerationKeyStandard &Microbiological
3  information.Organism code}}, <attr name='organismCode'>?</attr>}
4
5  {?CreateXMLTagWithValue {?FormatDecimal &Microbiological
6  information.Organism maximum value (UOM type)}, ".", 6}
7  , <attrQual name='organismMaximumValue' qual='{&Microbiological
8  information.Organism maximum value UOM (UOM type).Code (Unit system)}'>?</
9  attrQual>}
10  {?CreateXMLTagWithValue {?FormatDecimal &Microbiological
11  information.Organism reference value (UOM type)}, ".", 2}
12  , <attrQual name='organismReferenceValue' qual='{&Microbiological
13  information.Organism reference value UOM (UOM type).Code (Unit system)}'>?
14  </attrQual>}
15  {?CreateXMLTagWithValue {?FormatDecimal &Microbiological
16  information.Organism warning value (UOM type)}, ".", 2}
17  , <attrQual name='organismWarningValue' qual='{&Microbiological
18  information.Organism warning value UOM (UOM type).Code (Unit system)}'>?</
19  attrQual>}
20  </row>

```

New IM export template, call of "Collect Microbiological information" sub-module ("Items" main export module)

```

1  {$Collect Microbiological information}

```

New IM export template, content of "Microbiological information" sub-module

```

1  {?ValueSet "tmpOrganismCode", {?EnumerationKeyStandard {&Microbiological
   information.Organism code}}}}
2  <row>
3  {?CreateXMLTagWithValue {?ValueGet "tmpOrganismCode"}, <attr name="organism
   mCode">?</attr>}
4
5  {?CreateXMLTagWithContent {?ValueGet {?ValueGet "tmpOrganismCode"}
   _maxValue}, "<attrQualMany name=\"organismMaximumValue\">?</attrQualMany>"
   }
6  {?CreateXMLTagWithContent {?ValueGet {?ValueGet "tmpOrganismCode"}
   _refValue}, "<attrQualMany name=\"organismReferenceValue\">?</attrQualMany>"
   }
7  {?CreateXMLTagWithContent {?ValueGet {?ValueGet "tmpOrganismCode"}
   _warnValue}, "<attrQualMany name=\"organismWarningValue\">?</attrQualMany>"
   }
8
9  </row>
10
11  {!Reset variables}
12  {?ValueSet "tmpOrganismCode", ""}
13  {?ValueSet {?ValueGet "tmpOrganismCode"}_maxValue, ""}
14  {?ValueSet {?ValueGet "tmpOrganismCode"}_refValue, ""}
15  {?ValueSet {?ValueGet "tmpOrganismCode"}_warnValue, ""}

```

New IM export template, content of "Collect Microbiological information" sub-module

```

1  {?ValueSet "tmpOrganismCode", {?EnumerationKeyStandard {&Microbiological
   detail information.Organism code}}}}
2  {!Organism maximum value}
3  {?ValueSet
4    {!name of variable to set}{?ValueGet "tmpOrganismCode"}_maxValue,
5    {!existing value}{?ValueGet {?ValueGet "tmpOrganismCode"}_maxValue}
6    {!add new value}{?CreateXMLTagWithValue {?FormatDecimal
   {&Microbiological detail information.Organism maximum value},.,6}, "<value
   qual=\"{&Microbiological detail information.Unit.Code (Unit system)}\">?
   </value>"}
7  }
8  {!Organism reference value}
9  {?ValueSet
10    {!name of variable to set}{?ValueGet "tmpOrganismCode"}_refValue,
11    {!existing value}{?ValueGet {?ValueGet "tmpOrganismCode"}_refValue}
12    {!add new value}{?CreateXMLTagWithValue {?FormatDecimal
   {&Microbiological detail information.Organism reference value},.,2},
   "<value qual=\"{&Microbiological detail information.Unit.Code (Unit
   system)}\">?</value>"}
13  }
14  {!Organism warning value}

```

```

15  {?ValueSet
16      {!name of variable to set}{?ValueGet "tmpOrganismCode"}_warnValue,
17      {!existing value}{?ValueGet {?ValueGet "tmpOrganismCode"}_warnValue}
18      {!add new value}{?CreateXMLTagWithValue {?FormatDecimal
    {&Microbiological detail information.Organism warning value},.,2}, "<value
    qual=\"{&Microbiological detail information.Unit.Code (Unit system)}\">?
    </value>"}
19  }

```

Flex attributes: physioChemicalProperties

Affected fields: "Physiochemical characteristic value UOM"

The field is no longer mandatory; if the UOM is not maintained the attribute qual="<unit>" won't be added to the XML tag. In addition, output of a second entry for second UOM type has been added to the corresponding export template module.

Previous IM export template, content of "Physiochemical information" sub-module

```

1  {?CreateXMLTagWithValue {?EnumerationKeyStandard {&Physiochemical
2  information.Physiochemical characteristic code}},
3  "<attrGroupMany name="physioChemicalProperties">
4  <row>
5  <attr name="physioChemicalCharacteristicCode">?</attr>
6  {?CreateXMLTagWithValue {?FormatDecimal {&Physiochemical
7  information.Physiochemical characteristic value (UOM type)}, ., 6},
8  "<attrQualMany name="physioChemicalCharacteristicValue">
9  <value qual="{&Physiochemical information.Physiochemical
10 characteristic value UOM (UOM type).Code (Unit system)}">?</value>
11 </attrQualMany>"}
12 </row>
13 </attrGroupMany>"
14 }

```

New IM export template, content of "Physiochemical information" sub-module

```

1  {?ValueSet "outputCode", {?EnumerationKeyStandard {&Physiochemical
2  information.Physiochemical characteristic code}}}
3  {?ValueSet "outputValueFirst", {?FormatDecimal {&Physiochemical
4  information.Physiochemical characteristic value (UOM type)}, ".", 6}}
5  {?ValueSet "outputValueSecond", {?FormatDecimal {&Physiochemical
6  information.Physiochemical characteristic value (UOM type second)}, ".",
7  6}}
8  {?ValueSet "uomFirst", {&Physiochemical information.Physiochemical
9  characteristic value UOM (UOM type).Code (Unit system)}}
10 {?ValueSet "uomSecond", {&Physiochemical information.Physiochemical
11 characteristic value UOM (UOM type second).Code (Unit system)}}
12

```

```

7      {?IfNotEmptyThenNotEnc {?ValueGet "outputCode"}{?ValueGet
8      "outputValueFirst"}{?ValueGet "outputValueSecond"},
9      "<row>
10     {?CreateXMLTagWithValue {?ValueGet "outputCode"}, "<attr
11     name=\"physioChemicalCharacteristicCode\">{?</attr>"}
12     {?IfNotEmptyThenNotEnc {?ValueGet "outputValueFirst"}{?ValueGet
13     "outputValueSecond"}, "<attrQualOptMany
14     name=\"physioChemicalCharacteristicValue\">"}
15     {! value and unit for uom type}
16     {?ValueSet "uomQual", {?CreateXMLTagWithValue {?ValueGet "uomFirst"},
17     "qual=\"?\"\"}}
18     {?CreateXMLTagWithValue {?ValueGet "outputValueFirst"}, {?Concat "<value
19     ", {?ValueGet "uomQual"}, ">{?</value>"}
20     }
21     {! value and unit for uom type second, only if not equal to uom type and
22     uom differs}
23     {?IfNotEmptyThenNotEnc {?ValueGet "UOMTypeDiffers"}, {?Compare {?ValueGet
24     "uomFirst"}, {?ValueGet "uomSecond"}, "",
25     {?ValueSet "uomQual", {?CreateXMLTagWithValue {?ValueGet "uomSecond"},
26     "qual=\"?\"\"}}
27     {?CreateXMLTagWithValue {?ValueGet "outputValueSecond"}, {?Concat
28     "<value ", {?ValueGet "uomQual"}, ">{?</value>"}
29     }
30     }}
31     {?IfNotEmptyThenNotEnc {?ValueGet "outputValueFirst"}{?ValueGet
32     "outputValueSecond"}, "</attrQualOptMany>"}
33     </row>"
34     }


```

2.1.17 GDSN Migration Guide for version 3.1.12

2.1.17.1 Overview

This migration guide covers all changes we have made in Product 360 (versions 8.1.1.05 and 10.0.0.01) to support GDSN version 3.1.12, it refers to changes since GDSN version 3.1.8.

We have added some new fields with corresponding enumerations as well as added new entries to existing enumerations.

 There is no need to do any data migration.

In addition to that we have introduced some new data quality rule configurations to cover new validations. Our example export templates for submitting item data to the GDSN pools have been extended as well.

For more information regarding changes in individual GDSN versions, please have a look at the following chapters:

[GDSN 3.1.8 -> GDSN 3.1.9](#)(see page 286)

[GDSN 3.1.9 -> GDSN 3.1.10](#)(see page 291)

[GDSN 3.1.10 -> GDSN 3.1.11](#)(see page 294)

[GDSN 3.1.11 -> GDSN 3.1.12](#)(see page 296)

2.1.17.2 Product 360 data model changes

In this release we have nine new fields in context of GDSN or Food & Beverage in total, for two fields the descriptions have changed.

New data fields








The following table lists all new fields we have added.

GDSN attribute name	GDSN module	GDSN/ F&B	P360 field name	P360 entity name	Remarks
deliveryFrequencyCode	DeliveryPurchasingInformationModule	GDSN	Delivery frequency code	Delivery purchasing information	
isCertificateRequired	CertificationInformationModule	F&B	Is Certificate required?	Certification	Used in "Certification information" entity of items
isCertificateRequired	DietInformationModule	F&B	Is Certificate required?	Certification	Used in "Diets" entity of items ("Diets" is a sub-entity of "Diet related information")
fatPercentageInDryMatterMeasurementPrecisionCode	DairyFishMeatPoultryItemModule	F&B	Fat percentage in dry matter measurement precision code	Cheese information	"Cheese information" is a sub-entity of "Dairy fish meat poultry"
nutritionalScore	HealthRelatedInformationModule	F&B	Nutritional score	Health related information	
nutritionalValue	HealthRelatedInformationModule	F&B	Nutritional value	Health related information	
cannabisCBDTypeCode	HealthRelatedInformationModule	F&B	Cannabis CBD type	Health related information	


GDSN attribute name	GDSN module	GDSN/ F&B	P360 field name	P360 entity name	Remarks
nutritionalProgramIngredientTypeCode	HealthRelatedInformationModule	F&B	Nutritional program ingredient type	Health related information	
nutritionalProgramIngredientMeasurement	HealthRelatedInformationModule	F&B	Nutritional program ingredient value Nutritional program ingredient unit	Health related information UOM	As usual, we have implemented this attribute by two fields, the value and the corresponding unit. Note: As with other similar fields, we do not support the maintenance of multiple values per uom type.

2.1.17.3 Data quality rule and configuration changes

We have added new data quality rule configuration for two GDSN rules.

-  **Microbiological information (Standard GDSN) (Item)**
 -  Check 'Organism code' is not optional
-  **Health related information (Standard GDSN) (Item)**
 -  Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient type' populated
 -  Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient value' populated
 -  Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient type' populated
 -  Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient unit' populated

The following table lists all new rule configurations we have added.

 Please notice that the value <No code> for Organism code in Microbiological information is not allowed anymore. For compatibility reasons, we do not yet delete this value from the enumeration, but this will probably happen in a later version.

G D S N R u l e I D	GDSN Rule Description	GDSN Error Message	IDQ Rule Name	P360 Rule Configuration Name	P360 Rule Configuration Description	Error Message Code	P360 Error Message
1694	If any attributes in class microbiologicalInformation is provided, then the attribute microbiologicalOrganismCode must be provided.	If attribute XY (any attribute of class microbiologicalInformation) is provided, therefore microbiologicalOrganismCode SHALL be populated	Check_ConditionalNotEqual	Check 'Organism code' is not optional	Checks that 'Organism code' is not optional if in use. Optional value '<No code>' is not allowed	GD SN 1694	Value 'No code' is not allowed for 'Organism code'
1699	nutritionalProgramIngredientMeasurement shall only be used if nutritionalProgramIngredientTypeCode is used.	nutritionalProgramIngredientType code is not used.	Check_IfNotEmptyConditionNot Empty	Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient type' populated	Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient type' populated	GD SN 1699	Items with 'Nutritional program ingredient unit' or 'Nutritional program ingredient value' must have a 'Nutritional program ingredient type' populated

GDSN Rule ID	GDSN Rule Description	GDSN Error Message	IDQ Rule Name	P360 Rule Configuration Name	P360 Rule Configuration Description	Error Message Code	P360 Error Message
			Check_IfNotEmptyConditionNot Empty	Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient type' populated	Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient type' populated	GD SN 1699	Items with 'Nutritional program ingredient unit' or 'Nutritional program ingredient value' must have a 'Nutritional program ingredient type' populated
			Check_IfNotEmptyConditionNot Empty	Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient value' populated	Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient value' populated.	Ext 030	Items with 'Nutritional program ingredient unit' must have a 'Nutritional program ingredient value' populated

GDSN Rule ID	GDSN Rule Description	GDSN Error Message	IDQ Rule Name	P360 Rule Configuration Name	P360 Rule Configuration Description	Error Message Code	P360 Error Message
			Check_IfNotEmptyConditionNot Empty	Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient unit' populated	Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient unit' populated.	Ext 029	Items with 'Nutritional program ingredient value' must have a 'Nutritional program ingredient unit' populated

2.1.17.4 Changes in data source export templates

If you use the provided default "Catalog request item" (IM) or "Catalog item notification" (DSE) export template you should replace it by the new one. In case this is not possible because of customizations, you'll find a detailed description of changes below.

New XSD file set

A new XSD file set has been provided for GDSN version 3.1.12. All old XSD files used for export post steps have to be deleted, then all new XSD files must be uploaded. Export templates do not have to be adapted due to the changed XSD files.

DSE export template changes

The only changed export template for DSE is `CIN_CatalogItemNotification.ext`. The following fields have been added to that template.

Delivery frequency code

Export template module: Delivery purchasing information: Collect distribution details

Changes: line 7 has been added

Delivery purchasing information: Collect distribution details	
1	< distributionDetails >
2	{?CreateXMLTagWithValue {?EnumerationKeyStandard {&Distribution details.Distribution method code}}, "< distributionMethodCode >?</ distributionMethodCode >"}
3	{?CreateXMLTagWithValue {?EnumerationKey {&Distribution details.Is distribution method primary}}, "< isDistributionMethodPrimary >?</ isDistributionMethodPrimary >"}
4	{?CreateXMLTagWithValue {?FormatDecimal {&Distribution details.Ordering lead time (UOM type)}}, ".", 6}, "< orderingLeadTime measurementUnitCode=\"{&Distribution details.Ordering lead time UOM (UOM type).Code (Unit system)}\">?</ orderingLeadTime >"}
5	{?CreateXMLTagWithValue {?EnumerationKey {&Distribution details.Delivery frequency code}}, "< deliveryFrequencyCode >?</ deliveryFrequencyCode >"}
6	</ distributionDetails >

Is certificate required? (Diets)

Export template module: Diet related information: Collect diets

Changes: lines 20 and 21 have been added

Diet related information: Collect diets	
1	{?CreateXMLTagWithValue {&Diets.Certification.Certification standard}
2	, "< certificationStandard >?</ certificationStandard >"}
3	{?CreateXMLTagWithValue {?EnumerationKey {&Diets.Certification.Is certificate required?}}
4	, "< isCertificateRequired >?</ isCertificateRequired >"}
5	
6	{?IfEmptyThenNotEnc "", {?ValueGet {?ValueGet "tmpCertifications"}}}
7	</ dietCertification >
8	</ dietTypeInformation >

Fat percentage in dry matter measurement precision code

Export template module: Dairy fish meat poultry

Changes:

- line11: added new field "Fat percentage in dry matter measurement precision code" as part of check

- lines 30, 31: added new field, output the code of the maintained value

Dairy fish meat poultry

```

1      {?IfNotEmptyThenNotEnc {&Dairy fish meat poultry.Cheese maturation
period description (Language)}}{&Dairy fish meat poultry.Cheese maturation
container process type}}{&Dairy fish meat poultry.Fat in dry matter [%]}
{&Dairy fish meat poultry.Is rind edible}}{&Dairy fish meat poultry.Rennet
type code}}{&Dairy fish meat poultry.Ripening time period (UOM type)}
{&Dairy fish meat poultry.Surface of cheese at end of ripening}}{&Dairy
fish meat poultry.Fat percentage in dry matter measurement precision code}
2      , {?Concat "<cheeseInformation>"
3      , {?CreateXMLTagWithValue {&Dairy fish meat poultry.Cheese
maturation period description (Language)}
4      , "<cheeseMaturationPeriodDescription languageCode=\"{?
GDSNEnumerationCode {&Dairy fish meat poultry.Language (Language)},
"GDSN"}\">?</cheeseMaturationPeriodDescription>"}
5      , {?IfNotEmptyThen {?ValueGet "LanguageDiffs"}, {?
CreateXMLTagWithValue {&Dairy fish meat poultry.Cheese maturation period
description (Language second)}
6      , "<cheeseMaturationPeriodDescription languageCode=\"{?
GDSNEnumerationCode {&Dairy fish meat poultry.Language (Language second)},
"GDSN"}\">?</cheeseMaturationPeriodDescription>"}
7      , {?IfEmptyThenNotEnc "", {?SplitKeywords {?EnumerationKey {&Dairy
fish meat poultry.Cheese maturation container process type}}
8      , "<cheeseMaturationProcessContainerTypeCode>", "</
cheeseMaturationProcessContainerTypeCode>"}
9      , {?CreateXMLTagWithValue {?FormatDecimal {&Dairy fish meat
poultry.Fat in dry matter [%]}}, ".", 2}
10     , "<fatPercentageInDryMatter>?</fatPercentageInDryMatter>"}
11     , {?CreateXMLTagWithValue {?EnumerationKey {&Dairy fish meat
poultry.Is rind edible}}
12     , "<isRindEdible>?</isRindEdible>"}
13     , {?CreateXMLTagWithValue {?EnumerationKey {&Dairy fish meat
poultry.Rennet type code}}
14     , "<rennetTypeCode>?</rennetTypeCode>"}
15     {!No second UOM allowed}
16     , {?CreateXMLTagWithValue {?FormatDecimal {&Dairy fish meat
poultry.Ripening time period (UOM type)}}, ".", 6}
17     , "<ripeningTimePeriod measurementUnitCode=\"{&Dairy fish meat
poultry.Ripening time period UOM (UOM type).Code (Unit system)}\">?</
ripeningTimePeriod>"}
18     , {?CreateXMLTagWithValue {?EnumerationKey {&Dairy fish meat
poultry.Surface of cheese at end of ripening}}
19     , "<surfaceOfCheeseAtEndOfRipeningCode>?</
surfaceOfCheeseAtEndOfRipeningCode>"}
20     , {?CreateXMLTagWithValue {?EnumerationKey {&Dairy fish meat
poultry.Fat percentage in dry matter measurement precision code}}
21     , "<fatPercentageInDryMatterMeasurementPrecisionCode>?</
fatPercentageInDryMatterMeasurementPrecisionCode>"}

```

22

, "</cheeseInformation>"}}

IM export template changes

The only changed export template for Item Management is CR_CatalogueRequest Item.ext. The following fields have been added to that template.

Delivery frequency code

Export template module: Primary delivery information

Changes: lines 11 and 12 have been added

Primary delivery information

```

1  {?IfNotEmptyThenNotEnc {&Distribution details.Ordering lead time},
2  "<attrQualMany name="orderingLeadTime">
3  {?CreateXMLTagWithValue {?FormatDecimal {&Distribution details.Ordering
4  lead time},".", "6"},}
5  , "<value qual="{&Distribution details.Ordering lead time UOM.Code (Unit
6  system)}">?</value>"}
7  </attrQualMany>"
8  }
9  {?CreateXMLTagWithValue {?EnumerationKey {&Distribution details.Delivery
frequency code}}
, "<attr name=\"deliveryFrequencyCode\">?</attr>"}
</row>

```

Is Certificate required? (Item certifications)

Export template module: Certifications

Changes: line 8 has been added

Certifications

```

1  {?CreateXMLTagWithValue {&Certification
information.Certification.Certification organisation.Name}, "<attr
2  name=\"certificationAgency\">?</attr>"}
3  {?CreateXMLTagWithValue {&Certification
information.Certification.Certification organisation.GLN}, "<attr
name=\"certificationOrganisationIdentifier\">?</attr>"}
4  {?CreateXMLTagWithValue {&Certification
information.Certification.Certification standard}, "<attr
name=\"certificationStandard\">?</attr>"}

```


4	<pre>{?CreateXMLTagWithValue {?EnumerationKey {&Certification information.Certification.Is certificate required?}}, "<attr name=\"isCertificateRequired\">?</attr>"}</pre>
---	--

Is Certificate required? (Diets)

Export template module: Collect diets

Changes: line 23 has been added

Collect diets

1	<pre>{?CreateXMLTagWithContent {?ValueGet {?ValueGet "tmpCertifications"}}, "<attrGroupMany name=\"certification\">?</attrGroupMany>"}</pre>
2	
3	
4	
5	

```

{!Reset variable}{?ValueSet {?ValueGet "tmpCertifications"}, ""}
{?CreateXMLTagWithValue {&Diets.Certification (Diet type code
selectable).Is certificate required?}, "<attr
name=\"isCertificateRequired\">?</attr>"}
```

</row>

Fat percentage in dry matter measurement precision code

Export template module: Dairy fish meat poultry

Changes: lines 19 and 20 have been added

Dairy fish meat poultry

1	<pre>{?CreateXMLTagWithValue {?FormatDecimal {&Dairy fish meat poultry.Ripening time period},.,6}, <attrQual name="ripeningTimePeriod" qual="{&Dairy fish meat poultry.Ripening time period UOM.Code (Unit system)}">?</attrQual>} {?CreateXMLTagWithValue {?EnumerationKey {&Dairy fish meat poultry.Surface of cheese at end of ripening}},<attr name="surfaceOfCheeseAtEndOfRipeningC ode">?</attr>} {?CreateXMLTagWithValue {?EnumerationKey {&Dairy fish meat poultry.Fat percentage in dry matter measurement precision code}} ,<attr name=\"fatPercentageInDryMatterMeasurementPrecisionCode\">?</ attr>"}</pre>
2	
3	
4	

Fields of health related information

Affected fields: Cannabis CBD type, Nutritional program ingredient type, Nutritional program ingredient unit, Nutritional program ingredient value, Nutritional score, Nutritional value

Export template module: Health related information

Changes: lines 31 - 43 have been added

Health related information

```

1  {{?CreateXMLTagWithContent {{?SplitKeywords {{?EnumerationKey {{&Health
2  related information.Nutritional program code}}}
3  , "<value>", "</value>"}
4  , "<attrMany name=\"nutritionalProgramCode\">?</attrMany>"}
5  {{?CreateXMLTagWithValue {{&Health related information.Nutritional score}},
6  "<attr name=\"nutritionalScore\">?</attr>"}
7  {{?CreateXMLTagWithValue {{?FormatDecimal {{&Health related
8  information.Nutritional value}}, ".", 6}
9  , "<attr name=\"nutritionalValue\">?</attr>"}
10 {{?CreateXMLTagWithValue {{?EnumerationKey {{&Health related
11 information.Cannabis CBD type}}}
12 , "<attr name=\"cannabisCBDTypeCode\">?</attr>"}
13 {{?CreateXMLTagWithValue {{?EnumerationKey {{&Health related
14 information.Nutritional program ingredient type}}}
15 , "<attr name=\"nutritionalProgramIngredientTypeCode\">?</attr>"}
16 {{?IfNotEmptyThenNotEnc {{&Health related information.Nutritional program
ingredient value (UOM type)}}
, "<attrQualMany name=\"nutritionalProgramIngredientMeasurement\">
{{?CreateXMLTagWithValue {{?FormatDecimal {{&Health related
information.Nutritional program ingredient value (UOM type)}}, ".", "6", }
, "<value qual=\"{{&Health related information.Nutritional program
ingredient unit (UOM type)}.Code (Unit system)}\">?</value>"}
</attrQualMany>"
}

```

Changes according to XSD change

The position of the field "Product form" has been changed from flex attribute to named attribute.

Export template module: Items

Changes:

- lines 53 and 54 have been removed
- line 37 has been added

Old content for Product form in module "Items"

```

1  <attrGroup name="gpcBrickAttributes">
2  <attr name="gpcBrickCode">{{Item.Structure group(s) (GPC
Structure).Structure group identifier}}</attr>
3  {{?CreateXMLTagWithContent {{ $Brick attributes}},
4  "<attrGroupMany name=\"gpcBrickAttributeValues\">?</attrGroupMany>"}
5  </attrGroup>
6  {{?CreateXMLTagWithContent {{?SplitKeywords {{&Item.Product form}}
7  , "<value>", "</value>", "true"}, <attrQualOptMany name="productForm">?
</attrQualOptMany>}

```

```

8      </flex>
9      </globalAttributes>

```

New content for Product form in module "Items"

```

1      {?CreateXMLTagWithValue {?FormatDecimal {&Item.Quantity of next level
2      items},".",0},
3      "<totalQuantityOfNextLowerTradeItem>?</totalQuantityOfNextLowerTradeItem>
4      "}
5      {?SplitKeywords {&Item.Product form}, "<productForm>", </productForm>}
6      {?CreateXMLTagWithValue {?CompareBooleanValue {&Item.Is base unit}, true,
7      false,""},<isBaseUnit>?</isBaseUnit>}

```

2.1.17.5 B2B

With the Update to GDSN 3.1.12 the XSDs of DSE/Atrify and IM changed. These are used by the data transformations in B2B.

If you are a **cloud customer**, you don't need to do anything. A new EC2 instance with an updated B2B will be created for you from a new AMI.

If you use an **on premise** installation of B2B, you need to execute the following steps to update your system:

1. In your Power Center installation find the folder DataTransformation/ServiceDB and make a backup of this folder.
2. Stop the GDSN workflows.
3. Stop Power Center, Data Exchange and MFT.
4. In the PIM_<version>_resources_gdsn.delta.zip of the GDSN Accelerator Package in the folder Common you will find a file called B2B_GDSN.zip.
5. Extract all files and open DT_Services folder.
6. Depending on whether you use GDSN as a
 - a. Data Source (e. g. manufacturer), copy the content of the DataSource folder to the DataTransformation/ServiceDB folder of your B2B installation.
 - b. Data Recipient (e. g. retailer), copy the content of the DataRecipient folder to the Data Transformations/ServiceDB folder of your B2B installation.
7. Start Power Center, Data Exchange and MFT.
8. Start the workflows.

2.1.18 GDSN Migration Guide for version 3.1.15

2.1.18.1 Product 360 data model changes

In this release we have 5 new fields, 4 changed fields and 5 moved fields in context of GDSN or Food & Beverage in total, for two fields the descriptions have changed.

New data fields

The following table lists all new fields we have added.

GDSN attribute name	GDSN module	GDSN/ F&B	P360 field name	P360 entity	Remarks
isDietTypeMarkedOnPackage	Diet information	F&B	ArticleDietList.IsDietTypeMarkedOnPackage	Diets	
expressedAsPartOf	Nutritional information	F&B	ArticleNutrientListQuantity.ExpressedAsPartOf	Nutrient list quantity	
nutritionalProgramDetail	Health related information	F&B	ArticleNutritionalProgramLang.NutritionalProgramDetail	Nutritional program - Language specific data	
claimMarkedOnPackage	Nutritional Claim	F&B	ArticleNutritionalClaim.ClaimMarkedOnPackage	Nutritional claims	

GDSN attribute name	GDSN module	GDSN/ F&B	P360 field name	P360 entity	Remarks
ingredientStatement	Ingredient information	F&B	ArticleIngredientStatement.IngredientStatement ArticleIngredientStatement.Language ArticleIngredientStatement.Sequence	Ingredient information - Ingredient statements	The existing "Ingredient statement" field has been renamed into "Ingredient statement, computed" and can still be used

Changed data fields

GDSN attribute name	GDSN module	GDSN/ F&B	P360 field name	P360 entity	Change
compulsoryAdditive LabelInformation	Health related information	F&B	ArticleHealthCareLang. CompulsoryAdditivesLabel	ArticleHealthCare	Fieldtype changed from ArticleDomainLangType.Std_Text1000_01 to ArticleDomainLangType.Std_Text5000_01. Length increased to 5000
additionalDescription	TradeItem Description	GDSN	GDSNTargetMarketExtensionLang. AdditionalDescription	Target market specific GDSN language data	Fieldtype changed from ArticleMarketExtensionLangType. Std_Text1000_01 to ArticleMarketExtensionLangType. Std_Text2G_01. Length increased to 2000.

GDSN attribute name	GDSN module	GDSN /F&B	P360 field name	P360 entity	Change
certificationIdentification	Diet information Certification Information	F&B	Certificate.Identification	Certificates	Length increased from 35 to 120
certificationStandard	Diet information Certification Information	F&B	Certification.Standard	Certification	Length shortened from 200 to 120



If you are using DSE, please adjust the max length of Certificate.Value to 120 in the custom area of the repository.

Moved data fields

The following table lists fields that have been moved to another entity

GDSN attribute name	GDSN module	GDSN/ F&B	P360 field name	P360 entity	Remarks
nutritionalProgramCode	Health related information	F&B	From: ArticleHealthCare. NutritionalProgramCode To: ArticleNutritionalProgram. NutritionalProgramCode	From: Health related information To: Nutritional program	since it is now a logical key in its own entity, it is no longer a multi value field
nutritionalScore	Health related information	F&B	From: ArticleHealthCare. NutritionalScore To: ArticleNutritionalProgram. NutritionalScore	From: Health related information To: Nutritional program	
nutritionalValue	Health related information	F&B	From: ArticleHealthCare. NutritionalValue To: ArticleNutritionalProgram. NutritionalValue	From: Health related information To: Nutritional program	

GDSN attribute name	GDSN module	GDSN/ F&B	P360 field name	P360 entity	Remarks
nutritionalProgramIngredientTypeCode	Health related information	F&B	From: ArticleHealthCare. NutritionalProgramIngredientTypeCode To: ArticleNutritionalProgramUOM. NutritionalProgramIngredientTypeCode	From: Health related information To: Nutritional program UOM	
nutrientProgramIngredientMeasurement	Health related information	F&B	From: ArticleHealthCareUOM. NutritionalProgramIngredientValue ArticleHealthCareUOM. NutritionalProgramIngredientUnit To: ArticleNutritionalProgramUOM. NutritionalProgramIngredientValue ArticleNutritionalProgramUOM. NutritionalProgramIngredientUnit	From: Health related information UOM To: Nutritional program UOM	

2.1.18.2 Units of measure

The following table lists units that have been added.

An empty category means that the corresponding unit is part of the overall list of GDSN units, units with a category are part of the corresponding special unit enumeration, e.g. time units.

Code	Name (EN)	Description (EN)	Category
/L	per litre (/L)	Per Litre [/L] (Unified code)	
/mL	per millilitre (/mL)	Per millilitre [/mL] (Unified Code)	
/mmol	per millimole (/mmol)	Per millimole [/mmol] (Unified code)	
[APL'U]	IgA phospholipid units [APL'U]	IgA phospholipid units - biologic activity of anticardiolipin [APL'U] (Unified code)	
[beth'U]	Bethesda unit [beth'U]	Bethesda unit - biologic activity of factor VIII inhibitor [beth'U] (Unified Code)	
[GPL'U]	IgG phospholipid units [GPL'U]	IgG phospholipid units - biologic activity of anticardiolipin IgG [GPL'U] (Unified Code)	Volume
[HPF]	per high power field [HPF]	Per high power field - view area in microscope [/HPF] (Unified Code)	Energy
[iU]/d	international unit per day [iU/d]	International unit per day [iU/d] (Unified Code)	Time
[iU]/L	international unit per litre [iU/L]	International unit per litre [iU/L] (Unified Code)	

Code	Name (EN)	Description (EN)	Category
[iU]/mL	international unit per millilitre [iU/mL]	International unit per millilitre [iU/mL] (Unified Code)	Time
[LPF]	per low power field [LPF]	Per low power field - view area in microscope [/ [LPF]] (Unified Code)	Energy
[MPL'U]/L	IgM phospholipid units [MPL'U] per litre	IgM phospholipid units -biologic activity of anticardiolipin IgM – MPL unit per litre [[MPL'U]/L] (Unified code)	
B	Bel - level [B]	Bel - level [B] (Unified code)	
diop	dioptr - refraction of a lens [diop]	Dioptr - refraction of a lens [diop] (Unified code)	Mass
fmol/L	femtomole per litre [fmol/L]	Femtomole per litre [fmol/L] (Unified code)	Mass
g(48.h)	gram per 48 hour [g/48 hr]	Gram per 48 hour [g/48 hr] (Unified code)	Time
g/(12.h)	gram per 12 hour [g/12 hr]	Gram per 12 hour [g/12 hr] (Unified code)	Time
g/(4.h)	gram per 4 hour [g/4 hr]	Gram per 4 hour [g/4 hr] (Unified code)	Time
g/(6.h)	gram per 6 hour [g/6 hr]	Gram per 6 hour [g/6 hr] (Unified code)	Time
g/(72.h)	gram per 72 hour [g/72 hr]	Gram per 72 hour [g/72 hr] (Unified code)	Time
g/d	gram per day [g/d]	Gram per day [g/d] (Unified code)	Time
g/g	gram per gram [g/g]	Gram per gram [g/g] (Unified code)	

Code	Name (EN)	Description (EN)	Category
kat	katal - catalytic activity [kat]	Katal - catalytic activity [kat] (Unified code)	
kU/L	kilo unit per litre [kU/L]	Kilo unit per litre [kU/L] (Unified code)	
L/L	litre per litre [L/L]	Litre per litre [L/L] (Unified code)	
m[iU]/L	milliinternational units per litre [m[iU]/L]	Milliinternational units per litre [m[iU]/L] (Unified code)	
mg/(12.h)	milligram per 12 hour [mg/12 hr]	Milligram per 12 hour [mg/12 hr] (Unified code)	Time
mg/d	milligram per day [mg/d]	Milligram per day [mg/d] (Unified Code)	
mg/g	milligram per gram [mg/g]	Milligram per gram [mg/g] (Unified code)	
mg/L	milligram per litre [mg/L]	Milligram per litre [mg/L] (Unified code)	
mg/mg	milligram per milligram [mg/mg]	Milligram per milligram [mg/mg] (Unified code)	
mg/mL	milligram per millilitre [mg/mL]	Milligram per millilitre [mg/mL] (Unified code)	
mL/(10.h)	millilitre per 10 hour [mL/10 hr]	Millilitre per 10 hour [mL/10 hr] (Unified code)	Time
mL/(12.h)	millilitre per 12 hour [mL/12 hr]	Millilitre per 12 hour [mL/12 hr] (Unified code)	Time
mL/(2.h)	millilitre per 2 hour [mL/2 hr]	Millilitre per 2 hour [mL/2 hr] (Unified code)	Time
mL/(4.h)	millilitre per 4 hour [mL/4 hr]	Millilitre per 4 hour [mL/4 hr] (Unified code)	Time
mL/(5.h)	millilitre per 5 hour [mL/5 hr]	Millilitre per 5 hour [mL/5 hr] (Unified code)	Time

Code	Name (EN)	Description (EN)	Category
mL/(6.h)	millilitre per 6 hour [mL/6 hr]	Millilitre per 6 hour [mL/6 hr] (Unified code)	Time
mL/(72.h)	millilitre per 72 hour [mL/72 hr]	Millilitre per 72 hour [mL/72 hr] (Unified code)	Time
mL/(8.h)	millilitre per 8 hour [mL/8 hr]	Millilitre per 8 hour [mL/8 hr] (Unified Code)	Time
mL/d	millilitre per day [mL/d]	Millilitre per day [mL/d] (Unified Code)	Time
mL/h	millilitre per hour [mL/hr]	Millilitre per hour [mL/hr] (Unified Code)	Time
mmol/(12.h)	millimole per 12 hour [mmol/12 hr]	Millimole per 12 hour [mmol/12 hr] (Unified code)	Time
mmol/(5.h)	millimole per 5 hour [mmol/5 hr]	Millimole per 5 hour [mmol/5 hr] (Unified code)	Time
mmol/(6.h)	millimole per 6 hour [mmol/6 hr]	Millimole per 6 hour [mmol/6 hr] (Unified code)	Time
mmol/d	millimole per day [mmol/d]	Millimole per day [mmol/d] (Unified code)	Time
mmol/g	millimole per gram [mmol/g]	Millimole per gram [mmol/g] (Unified code)	
mmol/kg[H2O]	millimole per kilogram [H2O] [mmol/kg Water]	Millimole per kilogram [H2O] [mmol/kg Water] (Unified code)	
mmol/L	millimole per litre [mmol/L]	Millimole per litre [mmol/L] (Unified code)	
mmol/mmol	millimole per millimole [mmol/mmol]	Millimole per millimole [mmol/mmol] (Unified code)	
mU	milliUnit [mU]	MilliUnit [mU] (Unified code)	Mass

Code	Name (EN)	Description (EN)	Category
mU/L	milliUnit per litre [mU/L]	MilliUnit per litre [mU/L] (Unified code)	
ng/d	nanogram per day [ng/d]	Nanogram per day [ng/d] (Unified code)	Time
ng/g	nanogram per gram [ng/g]	Nanogram per gram [ng/g] (Unified code)	
ng/L	nanogram per litre [ng/L]	Nanogram per litre [ng/L] (Unified code)	
ng/mL	nanogram per millilitre [ng/mL]	Nanogram per millilitre [ng/mL] (Unified code)	
nmol/d	nanomole per day [nmol/d]	Nanomole per day [nmol/d] (Unified code)	Time
nmol/g	nanomole per gram [nmol/g]	Nanomole per gram [nmol/g] (Unified code)	
nmol/h/mL	nanomole per hour per millilitre [nmol/hr/mL]	Nanomole per hour per millilitre [nmol/h/mL] (Unified code)	
nmol/L	nanomole per litre [nmol/L]	Nanomole per litre [nmol/L] (Unified code)	
nmol/mmol	nanomole per millimole [nmol/mmol]	Nanomole per millimole [nmol/mmol] (Unified code)	
nmol/nmol	nanomole per nanomole [nmol/nmol]	Nanomole per nanomole [nmol/nmol] (Unified code)	
pg	picogram [pg]	Picogram [pg] (Unified code)	Mass
pg/mL	picogram per millilitre [pg/mL]	Picogram per millilitre [pg/mL] (Unified code)	
pmol/d	picomole per day [pmol/d]	Picomole per day [pmol/d] (Unified code)	Time

Code	Name (EN)	Description (EN)	Category
pmol/g	picomole per gram [pmol/g]	Picomole per gram [pmol/g] (Unified code)	
pmol/h/mg	picomole per hour per milligram [pmol/hr/mg]	Picomole per hour per milligram [pmol/hr/mg] (Unified code)	Time
pmol/h/mL	picomole per hour per millilitre [pmol/hr/mL]	Picomole per hour per millilitre [pmol/hr/mL] (Unified code)	
pmol/L	picomole per litre [pmol/L]	Picomole per litre [pmol/L] (Unified code)	
pmol/mmol	picomole per millimole [pmol/mmol]	Picomole per millimole [pmol/mmol] (Unified code)	
U/(12.h)	unit per 12 hour [U/12 hr]	Unit per 12 hour [U/12 hr] (Unified code)	Time
U/(2.h)	unit per 2 hour [U/2 hr]	Unit per 2 hour [U/2 hr] (Unified code)	Time
U/d	unit per day [U/d]	Unit per day [U/d] (Unified code)	Time
U/g	unit per gram [U/g]	Unit per gram [U/g] (Unified code)	-
U/h	unit per 1 hour [U/hr]	Unit per 1 hour [U/hr] (Unified code)	Time
U/kg	unit per kilogram [U/kg]	Unit per kilogram [U/kg] (Unified code)	
U/mL	unit per millilitre [U/mL]	Unit per millilitre [U/mL] (Unified code)	
ug/d	microgram per day [ug/d] aka [mcg/d]	Microgram per day [ug/d] (Unified code) aka [mcg/d]	Time
ug/g	microgram per gram [ug/g] aka [mcg/g]	Microgram per gram [ug/g] (Unified code) aka [mcg/g]	

Code	Name (EN)	Description (EN)	Category
ug/L	microgram per litre [ug/L] aka [mcg/L]	Microgram per litre [ug/L] (Unified code) aka [mcg/L]	
ug/min	microgram per minute [ug/min] aka [mcg/min]	Microgram per minute [ug/min] (Unified code) aka [mcg/min]	Time
ug/mL	microgram per millilitre [ug/mL] aka [mcg/mL]	Microgram per millilitre [ug/mL] (Unified code) aka [mcg/mL]	
um/s	micrometre per second [um/s]	Micrometre per second [um/s] (Unified code)	Time
umol/(2.h)	micromole per 2 hour [umol/2 hr]	Micromole per 2 hour [umol/2 hr] (Unified code)	Time
umol/d	micromole per day [umol/d]	Micromole per day [umol/d] (Unifed Code)	Time
umol/g	micromole per gram [umol/g]	Micromole per gram [umol/g] (Unified code)	
umol/L	micromole per litre [umol/L]	Micromole per litre [umol/L] (Unified code)	
umol/min/L	enzyme unit [micromoles/minute] per litre [umol/min/L]	Enzyme unit [micromoles/minute] per litre [umol/min/L] (Unified code)	
umol/mmol	micromole per millimole [umol/mmol]	Micromole per millimole [umol/mmol] (Unified code)	
umol/umol	micromole per micromole [umol/umol]	Micromole per micromole [umol/umol] (Unified code)	
g/L	gram per litre [g/L]	Gram per litre [g/L] (Unified code)	
mm[Hg]	millimetre of mercury [pressure]	Millimetre of mercury [pressure] (Unified code)	Pressure

Code	Name (EN)	Description (EN)	Category
u	unified atomic mass unit [u]	Unified atomic mass unit [u] (Unified code)	Mass
cd	Candela	Candela [cd] (Unified code)	
C	Coulomb	Coulomb - electric charge [C] (Unifed Code)	Energy
eV	electronvolt	Electronvolt - energy [eV] (Unifed code)	Energy
F	Farad	Farad - electric capacitance [F] (unified code)	Energy
Gy	Gray [Gy]	Gray - energy dose [Gy] (Unified Code)	Energy
har	Hecto - Hectare [ha]	Hectare: Hecto-are and hectare are equivalent units [har] (Unified code)	Mass
H	Henry [H]	Henry - inductance [H] (Unified Code)	
mL/min	millilitre per minute	Millilitre per minute [mL/min] (Unified Code)	Time
mL/s	millilitre per second	Millilitre per second [mL/s] (Unified Code)	Time
mmol/kg	millimole per kilogram [mmol/kg]	Millimole per kilogram [mmol/kg] (Unified code)	
Np	Neper	Neper - level [Np] (Unified code)	
S	Siemens - electric conductance [S]	Siemens - electric conductance [S] (Unified code)	
Sv	Sievert [Sv]	Sievert - dose equivalent [Sv] (Unified code)	

Code	Name (EN)	Description (EN)	Category
T	Tesla [T]	Tesla - magnetic flux density [T] (Unified code)	
D53	Watt per metre Kelvin	Specifies the thermal conductivity (lambda value): a measure of the product's ability to conduct heat.	Energy
Wb	Weber - magnetic flux [Wb]	Weber - magnetic flux [Wb] (Unified code)	

2.1.18.3 Data quality rule and configuration changes

The following table lists all rule configurations we have added, changed or deleted.

GDSN Rule ID	GDSN Rule	P360 Rule Configuration	P360 Error Message	Remarks
1699	<p><i>Description:</i> nutritionalProgramIngredientMeasurement shall only be used if nutritionalProgramIngredientTypeCode is used.</p> <p><i>Error message:</i> nutritionalProgramIngredientTypecode is not used</p>	<p><i>Name:</i> Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient type' populated</p> <p><i>Description:</i> Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient type' populated</p> <p><i>IDQ rule name:</i> Check_IfNotEmptyConditionNotEmpty</p>	<p><i>Code:</i> GDSN1699</p> <p><i>Message:</i> Items with 'Nutritional program ingredient unit' or 'Nutritional program ingredient value' must have a 'Nutritional program ingredient type' populated</p>	Deleted because nutritionalProgramIngredientTypeCode is now a logical key ensuring this rule.
1699	<p><i>Description:</i> nutritionalProgramIngredientMeasurement shall only be used if nutritionalProgramIngredientTypeCode is used.</p>	<p><i>Name:</i> Check items with 'Nutritional program ingredient value' have a 'Nutritional porgam ingredient type' populated</p>	<p><i>Code:</i> GDSN1699</p>	Deleted because nutritionalProgramIngredientTypeCode is now a logical key ensuring this rule.

GDSN Rule ID	GDSN Rule	P360 Rule Configuration	P360 Error Message	Remarks
	<p><i>Error message:</i> nutritionalProgramIngredientTypecode is not used</p>	<p><i>Description:</i> Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient type' populated</p> <p><i>IDQ rule name:</i> Check_IfNotEmptyConditionNotEmpty</p>	<p><i>Message:</i> Items with 'Nutritional program ingredient unit' or 'Nutritional program ingredient value' must have a 'Nutritional program ingredient type' populated</p>	
-	-	<p><i>Name:</i> Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient value' populated</p> <p><i>Description:</i> Check items with 'Nutritional program ingredient unit' have a 'Nutritional program ingredient value' populated</p> <p><i>IDQ rule name:</i> Check_IfNotEmptyConditionNotEmpty</p>	<p>Code: Ext030</p> <p><i>Message:</i> Items with 'Nutritional program ingredient unit' must have a 'Nutritional program ingredient value' populated</p>	<p>Adjusted changed field (ArticleHealthCareUOM.NutritionalProgramIngredientUnit → ArticleNutritionalProgramUOM.NutritionalProgramIngredientUnit)</p>

GDSN Rule ID	GDSN Rule	P360 Rule Configuration	P360 Error Message	Remarks
-	-	<p><i>Name:</i> Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient unit' populated</p> <p><i>Description:</i> Check items with 'Nutritional program ingredient value' have a 'Nutritional program ingredient unit' populated</p> <p><i>IDQ rule name:</i> Check_IfNotEmptyConditionNotEmpty</p>	<p><i>Code:</i> Ext029</p> <p><i>Message:</i> Items with 'Nutritional program ingredient value' must have a 'Nutritional program ingredient unit' populated</p>	Adjusted changed field (ArticleHealthCareUOM.NutritionalProgramIngredientUnit → ArticleNutritionalProgramUOM.NutritionalProgramIngredientUnit)
1698	<p><i>Description:</i> If nutritionalProgramCode equals '8' (Nutri-Score), then its related value in nutritionalScore shall equal ('A', 'B', 'C','D', 'E' or 'EXEMPT')</p> <p><i>Error message:</i> Acceptable values for Nutri-Score program are ('A', 'B', 'C','D', 'E' or 'EXEMPT')</p>	<p><i>Name:</i> check items with 'Nutritional program' equals 'Nutriscore', have a 'Nutritional score' in 'A', 'B', 'C', 'D', 'E' or 'EXEMPT'</p> <p><i>Description:</i> Checks that if 'Nutritional program' equals 'Nutriscore' that the related 'Nutritional score' has a valid value ('A',</p>	<p><i>Code:</i> GDSN1698</p> <p><i>Message:</i> Acceptable values for Nutri-Score program are ('A', 'B', 'C','D', 'E' or 'EXEMPT')</p>	New rule configuration

GDSN Rule ID	GDSN Rule	P360 Rule Configuration	P360 Error Message	Remarks
		'B', 'C', 'D', 'E', 'EXEMPT') <i>IDQ rule name:</i> Check_ConditionalIsInList		
457	<p><i>Description:</i> isTradeltemAnInvoiceUnit must equal 'true' for at least one trade item in a Catalogue Item Notification Message.</p> <p><i>Error message:</i> One or more of the GTINs in each hierarchy must be marked as an INVOICE UNIT</p>	<p><i>Name:</i> Check 'Is invoice unit' is true for at least one item in a packaging hierarchy (TM: US)</p> <p><i>Description:</i> Checks if 'Is invoice unit' is true for at least one item in a packaging hierarchy for US target market</p> <p><i>IDQ rule name:</i> Check_EqualsWithinHierarchy</p>	<p><i>Code:</i> GDSN234</p> <p><i>Message:</i> 'Is invoice unit' must be true for at least one item in a packaging hierarchy</p>	Deleted rule (according to GDSN validation documentation)
1705	<p><i>Description:</i> If dietTypeCode equals 'PESCATARIAN' and dietTypeSubcode is used, then</p>	<p><i>Name:</i> Check 'Diet subtype' has a valid value for 'Diet type' Pescatarian</p>	<i>Code:</i> GDSN1705	

GDSN Rule ID	GDSN Rule	P360 Rule Configuration	P360 Error Message	Remarks
	<p>dietTypeSubcode SHALL be a value in ('PESCA', 'LACTO_OVO_PESCA').</p> <p><i>Error message:</i> The dietTypeSubcodeCode is not valid for a Pescatarian diet.</p>	<p><i>Description:</i> Checks that diet subtype has a valid value when diet type is set to Pescatarian. Valid values for diet subtype are: Pesca, Lacto-Ovo-Pesca.</p> <p><i>IDQ rule name:</i> Check_ConditionallIsInList</p>	<p><i>Message:</i> The diet subtype is not valid for a Pescatarian diet</p>	
1706	<p><i>Description:</i> If dietTypeCode equals 'VEGETARIAN' and dietTypeSubcode is used and targetMarketCountryCode is not '840' (US), then dietTypeSubcode SHALL be a value in ('OVO', 'LACTO', 'LACTO_OVO').</p> <p><i>Error message:</i> The dietTypeSubcodeCode is not valid for a Vegetarian diet.</p>	<p><i>Name:</i> Check 'Diet subtype' has a valid value for 'Diet type' Vegetarian (TM: DE)</p> <p><i>Description:</i> Checks that 'Diet subtype' equals one of the provided values from the list 'Ovo,Lacto,Lacto-Ovo', if 'Diet type' is set to 'Vegetarian' and 'Target market' is not USA.</p>	<p><i>Code:</i> GDSN1706</p> <p><i>Message:</i> The 'Diet subtype' is not valid for a Vegetarian diet</p>	<p>Only added for target market 'Germany'.</p> <p>Please copy this configuration and adjust it to additional target markets you need to check.</p>

GDSN Rule ID	GDSN Rule	P360 Rule Configuration	P360 Error Message	Remarks
		<p><i>IDQ rule name:</i> Check_ConditionallIsInList</p> <p><i>Note:</i> This configuration checks only values for target market Germany, it should be copied and adjusted for all other needed target markets.</p>		
1707	<p><i>Description:</i> If dietTypeCode equals 'VEGETARIAN' and dietTypeSubcode is used and targetMarketCountryCode is '840' (US), then dietTypeSubcode SHALL be a value in ('OVO', 'LACTO', 'LACTO_OVO', 'PESCA', 'LACTO_OVO_PESCA').</p> <p><i>Error message:</i> The dietTypeSubcodeCode is not valid for a Vegetarian diet.</p>	<p><i>Name:</i> Check 'Diet subtype' has a valid value for 'Diet type' Vegetarian (TM: US)</p> <p><i>Description:</i> Checks that 'Diet subtype' has a valid value for target market USA when 'Diet type' is set to vegetarian. Valid values for diet subtype are: Ovo, Lacto, Lacto-Ovo, Pesca, Lacto-Ovo-Pesca.</p> <p><i>IDQ rule name:</i> Check_ConditionallIsInList</p>	<p><i>Code:</i> GDSN1707</p> <p><i>Message:</i> The 'Diet subtype' is not valid for a Vegetarian diet and target market USA</p>	

GDSN Rule ID	GDSN Rule	P360 Rule Configuration	P360 Error Message	Remarks
1708	<p><i>Description:</i> If dietTypeCode equals 'KOSHER' and dietTypeSubcode is used, then dietTypeSubcode SHALL be a value in ('MEAT', 'FISH', 'DAIRY', 'PAREVE', 'KOSHER_FOR_PASSOVER', 'MEAT_FOR_PASSOVER', 'FISH_FOR_PASSOVER', 'DAIRY_FOR_PASSOVER', 'PAREVE_FOR_PASSOVER', 'DE', 'MEVUSHAL', 'KOSHER_FOR_PASSOVER_MEVUSHAL').</p> <p><i>Error message:</i> The dietTypeSubcodeCode is not valid for a Kosher diet.</p>	<p><i>Name:</i> Check 'Diet subtype' has a valid value for 'Diet type' Kosher</p> <p><i>Description:</i> Checks that diet subtype has a valid value when diet type is set to Kosher. Valid values for diet subtype are: Meat, Fish, Dairy, Pareve, Kosher for Passover, Meat for Passover, Fish for passover, Dairy for passover, Parever for Passover, Kosher Dairy Equipment, Kosher Mevushal, Kosher for Passover-Mevushal.</p> <p><i>IDQ rule name:</i> Check_ConditionallIsInList</p>	<p><i>Code:</i> GDSN1708</p> <p><i>Message:</i> The 'Diet subtype' is not valid for a Kosher diet</p>	

2.1.18.4 Changes in data source export templates

New XSD file set

A new XSD file set has been provided for GDSN version 3.1.15. All old XSD files used for export post steps have to be deleted, then all new XSD files must be uploaded. Export templates do not have to be adapted due to the changed XSD files.

DSE export template changes

The only changed export template for DSE is `CIN_CatalogItemNotification.ext`. The following fields or modules have been added to that template.

Is diet type marked on package

Export template module: *Diet related information: Collect diets*

Changes: line 10 and 11 have been added

Diet related information: Collect diets	
1	<code>{?CreateXMLTagWithValue {?EnumerationKey {&Diets.Diet subtype}}}</code>
2	<code>, "<dietTypeSubcode>?</dietTypeSubcode>"}</code>
3	<code>{?CreateXMLTagWithValue {?EnumerationKey {&Diets.Is diet type marked on package}}}</code>
4	<code>, "<isDietTypeMarkedOnPackage>?</isDietTypeMarkedOnPackage>"}</code>

Expressed as part of

Export template module: *Nutritional information: Collect nutrients*

Changes: line 21 and 22 have been added

Nutritional information: Collect nutrients (by serving)	
1	<code>{?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Expressed as part of (By serving, UOM type)}}}</code>
2	<code>, "<expressedAsPartOf?</expressedAsPartOf>"}</code>
3	<code></nutrientDetail></code>
4	<code>}</code>
5	<code>}</code>

Changes: line 47 and 48 have been added

Nutritional information: Collect nutrients (by measure)	
1	<code>{?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Expressed as part of (By measure, UOM type)}}}</code>
2	<code>, "<expressedAsPartOf?</expressedAsPartOf>"}</code>
3	<code></nutrientDetail></code>
4	<code>}</code>
5	<code>}</code>

Claim marked on package

Export template module: *Nutritional claims*

Changes: lines 6-7 have been added

Nutritional claims	
1	<code>{?CreateXMLTagWithValue {?EnumerationKeyStandard {&Nutritional claims.Nutritional claim element code}}</code>
2	<code>, "<nutritionalClaimNutrientElementCode>?</nutritionalClaimNutrientElementCode>"}</code>
3	<code>{?CreateXMLTagWithValue {?EnumerationKey {&Nutritional claims.Claim marked on package}}</code>
4	<code>, "<claimMarkedOnPackage>?</claimMarkedOnPackage>"}</code>
5	<code></nutritionalClaimDetail></code>

Ingredient statement

Export template module: *Ingredient information: collect ingredient statements*

Changes: new module

Data type filter:

- *Language*: variables *Language* and *Language second*
- *Target market*: variable *Target market*

Ingredient information: collect ingredient statements	
1	<code>{?ValueSetLocal "ingredientStatements",</code>
2	<code>{?ValueGetLocal "ingredientStatements"}</code>
3	<code>{?CreateXMLTagWithValue {&Ingredient statements.Ingredient statement}</code>
4	<code>, "<ingredientStatement languageCode=\"{?GDSNEnumerationCode {&Ingredient statements.Language}, "GDSN"}\"</code>
5	<code>sequenceNumber=\"{?FormatDecimal {&Ingredient statements.Sequence}, "",0}\">?</ingredientStatement>"}</code>

6	}
---	---

Export template module: *Ingredient information*

Changes: line 4 replaces former lines 4-7

Ingredient information	
1	{?SplitKeywords {?EnumerationKey {&Ingredient information.Ingredient of concern code}}
2	, "<ingredientOfConcernCode>","</ingredientOfConcernCode>"}
3	
4	{?IfEmptyThenNotEnc ,{?ValueGetLocal "ingredientStatements"}{?ValueSetLocal "ingredientStatements",""}}
5	
6	{?CreateXMLTagWithValue {?FormatDecimal {&Ingredient information.Juice content [%]},".", "2" }
7	, "<juiceContentPercent>?</juiceContentPercent>"}

Export template module: *Item*

Changes: line 167 has been added

Items	
1	{\$Ingredient information: collect ingredient statements}
2	{?CreateXMLTagWithContent {?Concat {\$Ingredient information}, {\$Ingredient additives}, {\$Ingredients}}}
3	, "<food_and_beverage_ingredient:foodAndBeverageIngredientModule xsi:schemaLocation="urn:gs1:gdsn:food_and_beverage_ingredient:xsd:3 http://www.gdsregistry.org/3.1/schemas/gs1/gdsn/ FoodAndBeverageIngredientModule.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:food_and_beverage_ingredient="urn:gs1:gdsn:food_and_beverage_ingredient:xsd:3">
4	?

```

5      </food_and_beverage_ingredient:foodAndBeverageIngredientModule>"
6    }

```

Nutritional program

The output of the *Health related information module* was completely new added to the export template.

Export template modules:

- Health related information
- Health related information: Collect nutritional program ingredients
- Health related information: Collect nutritional program

Changes: new modules

Export template module: *Item*

Changes: lines 225-231 have been added

Items	
1	{ \$Dairy fish meat poultry: Collect content }
2	{ ?CreateXMLTagWithContent { \$Dairy fish meat poultry }
3	, "<dairy_fish_meat_poultry:dairyFishMeatPoultryItemModule xsi:schemaLocation="urn:gs1:gdsn:dairy_fish_meat_poultry:xsd:3 http://www.gdsregistry.org/3.1/schemas/gs1/gdsn/ DairyFishMeatPoultryItemModule.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:dairy_fish_meat_poultry="urn:gs1:gdsn:dairy_fish_meat_poultry:xsd:3">
4	?
5	</dairy_fish_meat_poultry:dairyFishMeatPoultryItemModule>"

```

6   }
7
8   {$Health related information: Collect nutritional program ingredients}
9   {$Health related information: Collect nutritional program}
10  {?CreateXMLTagWithContent {$Health related information}
11    , "<health_related_information:healthRelatedInformationModule
      xsi:schemaLocation='urn:gs1:gdsn:health_related_information:xsd:3 http://www.gdsregistry.org/3.1/schemas/gs1/gdsn/HealthRelatedInformationModule.xsd' xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
      xmlns:health_related_information='urn:gs1:gdsn:health_related_information:xsd:3'>
12    ?
13    </health_related_information:healthRelatedInformationModule>"
14  }

```

Nutrient basis quantity type: By Package or Container

Export template module: *Nutritional information: Collect nutrients*

Changes: lines 53-77 have been added

Nutritional information: Collect nutrients

```

1   {!Collect "By container"}
2   {?ValueSetLocal "ByContainer", "ByContainer_{&Nutrients.Preparation state}_{&Nutrients.Target market}"}
3   {?ValueSetLocal {?ValueGetLocal "ByContainer"}, {?ValueGetLocal {?ValueGetLocal "ByContainer"}}
4     {?IfNotEmptyThenNotEnc {&Nutrients.Percentage of daily intake (By Package or Container, UOM type)}
      {&Nutrients.Measurement precision (By Package or Container, UOM type)}{&Nutrients.Quantity contained (By Package or
      Container, UOM type)}{&Nutrients.Quantity contained (By Package or Container, UOM type second)}},
5   <nutrientDetail>
6     <nutrientTypeCode>{?EnumerationKey {&Nutrients.Nutrient type}}</nutrientTypeCode>
7     {!No second UOM allowed}

```

```

8      {?CreateXMLTagWithValue {?FormatDecimal {&Nutrients.Percentage of daily intake (By Package or Container, UOM type)},
9      ".", 2}
10      , "<dailyValueIntakePercent>?</dailyValueIntakePercent>"}
11      {!No second UOM allowed}
12      {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Measurement precision (By Package or Container, UOM type)}}
13      , "<measurementPrecisionCode>?</measurementPrecisionCode>"}
14      {?CreateXMLTagWithValue {?FormatDecimal {&Nutrients.Quantity contained (By Package or Container, UOM type)}, "." , "6"}
15      , "<quantityContained measurementUnitCode=\"{&Nutrients.Quantity contained UOM (By Package or Container, UOM
16      type).Code (Unit system)}\">?</quantityContained>"}
17      {?IfNotEmptyThenNotEnc {?ValueGet "UOMTypeDiffs"}
18      , {?Compare {&Nutrients.Quantity contained UOM (By Package or Container, UOM type second).Code (Unit system)},
19      {&Nutrients.Quantity contained UOM (By Package or Container, UOM type).Code (Unit system)}},
20      {?CreateXMLTagWithValue {?FormatDecimal {&Nutrients.Quantity contained (By Package or Container, UOM type
21      second)}}, "." , "6"}
22      , "<quantityContained measurementUnitCode=\"{&Nutrients.Quantity contained UOM (By Package or Container, UOM
23      type second).Code (Unit system)}\">?</quantityContained>"}
24      }
25      }
26      }
27      {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Expressed as part of (By measure, UOM type)}}
28      , "<expressedAsPartOf>?</expressedAsPartOf>"}
29      </nutrientDetail>
30      }
31      }

```

Export template module: *Nutritional information*

Changes: lines 1 and 5-31 have been added

Nutritional information

```

1  {?ValueSetLocal "ByContainer", "ByContainer_{&Nutritional information.Preparation state}_{&Nutritional
2  information.Target market}"}
3  {?ValueSetLocal "ByMeasure", "ByMeasure_{&Nutritional information.Preparation state}_{&Nutritional information.Target
4  market}"}
5  {?ValueSetLocal "ByServing", "ByServing_{&Nutritional information.Preparation state}_{&Nutritional information.Target
6  market}"}
7
8  {!BY_CONTAINER only export when there is data for "by container"}
9  {?IfNotEmptyThenNotEnc {?ValueGetLocal {?ValueGetLocal "ByContainer"}},
10 <nutrientHeader>
11   <preparationStateCode>{?EnumerationKey {&Nutritional information.Preparation state}}</preparationStateCode>
12   {?CreateXMLTagWithValue {&Nutritional information.Daily value intake reference (Language)}
13   , "<dailyValueIntakeReference languageCode=\"{?GDSNEnumerationCode {&Nutritional information.Language (Language)},
14   "GDSN"}\">?</dailyValueIntakeReference>"}
15   {?IfNotEmptyThenNotEnc {?ValueGet "LanguageDiffs"}
16   , {?CreateXMLTagWithValue {&Nutritional information.Daily value intake reference (Language second)}
17   , "<dailyValueIntakeReference languageCode=\"{?GDSNEnumerationCode {&Nutritional information.Language (Language
18   second)}\", "GDSN"}\">?</dailyValueIntakeReference>"}
19   }
20
21   <nutrientBasisQuantityTypeCode>BY_PACKAGE_OR_CONTAINER</nutrientBasisQuantityTypeCode>
22   {?CreateXMLTagWithValue {?FormatDecimal {&Nutritional information.Nutrient basis quantity (UOM type)}}, ".", "6"}
23   , "<nutrientBasisQuantity measurementUnitCode=\"{&Nutritional information.Nutrient basis quantity UOM (UOM type).Code
24   (Unit system)}\">?</nutrientBasisQuantity>"}
25   {?CreateXMLTagWithValue {?FormatDecimal {&Nutritional information.Serving size (UOM type)}},.,6,}
26   , "<servingSize measurementUnitCode=\"{&Nutritional information.Serving size UOM (UOM type).Code (Unit system)}\">?</
27   servingSize>"}
28   {?CreateXMLTagWithValue {&Nutritional information.Serving size description (Language)}}

```



```

22      , "<servingSizeDescription languageCode=\"{?GDSNEnumerationCode {&Nutritional information.Language (Language)}}, "GDSN"
23      }\>?</servingSizeDescription>"}
24      {?IfNotEmptyThenNotEnc {?ValueGet "LanguageDiffs"}
25      , {?CreateXMLTagWithValue {&Nutritional information.Serving size description (Language second)}
26      , "<servingSizeDescription languageCode=\"{?GDSNEnumerationCode {&Nutritional information.Language (Language
27      second}}}, "GDSN"}\>?</servingSizeDescription>"}
28      }
29      {?IfEmptyThenNotEnc "", {?ValueGetLocal {?ValueGetLocal "ByContainer"}}}
30      {!Reset variable}{?ValueSetLocal {?ValueGetLocal "ByContainer"}, ""}
31    </nutrientHeader>
32  }
33  {!BY_MEASURE only export when there is data for "by measure"}

```

IM export template changes

The only changed export template for Item Management is CR_CatalogueRequest Item.ext. The following fields or modules have been added to that template.

Is diet type marked on package

Export template module: *Collect diets*

Changes: line 11 and 12 have been added

Collect diets	
1	{?CreateXMLTagWithValue {?EnumerationKey {&Diets.Diet subtype}},

```

2  "<attr name="dietTypeSubcode">?</attr>"
3
4  {?CreateXMLTagWithValue {?EnumerationKey {&Diets.Is diet type marked on package}},
5  "<attr name="isDietTypeMarkedOnPackage">?</attr>"

```

Expressed as part of

Export template module: *Collect diets*

Changes: lines 17-19 have been added

Collect nutrients (by serving)

```

1  {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Measurement precision (UOM type, By serving)}}{?IfEmptyThen
   {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Measurement precision (UOM type, By serving)}},
   {?EnumerationKey {&Nutrients.Measurement precision (UOM type
2     second, By serving)}}}
3     , "<attr name="measurementPrecisionCode">?</attr>"
4   }
5   {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Expressed as part of (By serving, UOM type)}}{?IfEmptyThen
   {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Expressed as part of (By serving, UOM type)}},
   {?EnumerationKey {&Nutrients.Expressed as part of (By serving,
6     UOM type second)}}}
7     , "<attr name="expressedAsPartOf">?</attr>"
   }

```

Changes: lines 53 - 55 have been added

Collect nutrients (by measure)

```

1      {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Measurement precision (UOM type, By measure)}} {?IfEmptyThen
2      {&Nutrients.Measurement precision (UOM type, By measure)} ,{?EnumerationKey {&Nutrients.Measurement precision (UOM type
3      second, By measure)}}}
4      , "<attr name='measurementPrecisionCode'>?</attr>"
5      }
6      {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Expressed as part of (By measure, UOM type)}} {?IfEmptyThen
7      {&Nutrients.Expressed as part of (By measure, UOM type)}, {?EnumerationKey {&Nutrients.Expressed as part of (By measure,
      UOM type second)}}}
      , "<attr name='expressedAsPartOf'>?</attr>"
      }

```

Claim marked on package

Export template module: *Nutritional claim*

Changes: lines 7-8 have been added

Nutritional claim

```

1      {?CreateXMLTagWithValue {?EnumerationKey {&Nutritional claims.Claim Marked On Package}}
2      , "<attr name='claimMarkedOnPackage'>?</attr>"}
3      </row>

```

Ingredient statement

Export template module: *Collect Ingredient statement*

Changes: new module

Data type filter:

- *Language*: variables *Language* and *Language second*
- *Target market*: variable *Target market*

Sorting:

- *Ingredient statements.Sequence*, descending order

Data field validation:

- Field: *Ingredient statements.Ingredient statement*
- Validation: *May not be blank*
- Action on error: *Remove data record*
- Error classification: *Information*

Collect Ingredient statement

```

1  {?ValueSetLocal "newSequence", {?Compare {?ValueGetLocal "sequence"},{&Ingredient statements.Sequence},"","x"}}
2  {?ValueSetLocal "startRow", {?IfNotEmptyThen {?ValueGetLocal "newSequence"}, {?Compare 1,{?DatasetCounter},"","x"}}}
3  {?ValueSetLocal "sequence", {&Ingredient statements.Sequence}}
4
5  {?ValueSetLocal "ingredientStatements",
6    <value qual="\{"?GDSNEnumerationCode {&Ingredient statements.Language},"GDSN"}\ ">{&Ingredient statements.Ingredient
    statement}</value>

```

```

7      {?IfNotEmptyThen {?ValueGetLocal "newSequence"},
8      </attrQualMany>
9      <attr name="sequenceNumber">{?FormatDecimal {&Ingredient statements.Sequence},"",0}</attr>
10     </row>}
11     {?IfNotEmptyThen {?ValueGetLocal "startRow"},
12     <row>
13     <attrQualMany name="statement">}
14     {?ValueGetLocal "ingredientStatements"}
15     }

```

Export template module: *Ingredient information*

Changes: lines 4-5 replace former lines 3-10; line 10 has been added

Ingredient information

```

1      {?CreateXMLTagWithContent {?SplitKeywords {?EnumerationKey {&Ingredient information.Ingredient of concern
2      code}}, "<value>", "</value>"}
3      , "<attrMany name=\"ingredientOfConcernCode\">?</attrMany>"}
4      {?CreateXMLTagWithContent {?ValueGetLocal "ingredientStatements"},
5      "<attrGroupMany name=\"ingredientStatement\"><row><attrQualMany name=\"statement\">?</attrGroupMany>"}
6
7      {?CreateXMLTagWithValue {?FormatDecimal {&Ingredient information.Juice content percentage},".", "2" }
8      , "<attr name=\"juiceContentPercentage\">?</attr>"}
9
10     {!Reset}{?ValueSetLocal "sequence", ""}{?ValueSetLocal "ingredientStatements", ""}

```

Export template module: *Item*

Changes: line 128 has been added

Items	
1	<code>{Collect packaging material}</code>
2	<code>{Collect Ingredient statement}</code>
3	<code>{Collect Microbiological information}</code>

Nutritional program

Export template module: *Collect nutritional program ingredients*

Changes: new module

Sorting:

- *Nutritional program ingredients.Nutritional program*, ascending order
- *Nutritional program ingredients.Nutritional program ingredient type*, ascending order

Data type filter:

- *Target market*: variable *Target market*
- *UOM type*: variable *UOM type*

Collect nutritional program ingredients	
1	<code>{?ValueSetLocal {&Nutritional program ingredients.Nutritional program},</code>
2	<code>{?ValueGetLocal {&Nutritional program ingredients.Nutritional program}}</code>
3	<code>{?CreateXMLTagWithContent</code>
4	<code>{?CreateXMLTagWithValue {?FormatDecimal {&Nutritional program ingredients.Nutritional program ingredient value},".",6}</code>
	<code>,</code>

```

5      "<attrQualMany name='nutritionalProgramIngredientMeasurement'>
6      <value qual='\{"&Nutritional program ingredients.Nutritional program ingredient unit.Code (Unit system)}\}">?</
value>
7      </attrQualMany>"
8      {?CreateXMLTagWithValue {?EnumerationKey {&Nutritional program ingredients.Nutritional program ingredient type}},
9      "<attr name='nutritionalProgramIngredientTypeCode'>?</attr>"
10     , "<row>?</row>"}
11   }

```

Export template module: *Collect nutritional programs*

Changes: new module

Data type filter:

- *Target market*: variable *Target market*

Collect nutritional programs

```

1  {?ValueSetLocal "tmpNutritionalProgram", {?ValueGetLocal "tmpNutritionalProgram"}}
2  <row>
3  <attr name='nutritionalProgramCode'>{?EnumerationKey {&Nutritional programs.Nutritional program}}</attr>
4  {?CreateXMLTagWithContent
5  {?CreateXMLTagWithValue {&Nutritional programs.Nutritional program detail (Language)},
6  "<value qual='\{"?GDSNEnumerationCode {&Nutritional programs.Language (Language)},'GDSN'}\}">?</value>"}
7  {?IfNotEmptyThen {?ValueGet "LanguageDiffers"},
8  {?CreateXMLTagWithValue {&Nutritional programs.Nutritional program detail (Language second)},
9  "<value qual='\{"?GDSNEnumerationCode {&Nutritional programs.Language (Language second)},'GDSN'}\}">?</value>"}
10  }
11  , "<attrQualMany name='nutritionalProgramDetail'\}">?</attrQualMany>"
12  }

```

```

13      {?CreateXMLTagWithValue {&Nutritional programs.Nutritional score},
14          "<attr name=\"nutritionalScore\">?</attr>"}
15      {?CreateXMLTagWithValue {?FormatDecimal {&Nutritional programs.Nutritional value}, ".",6},
16          "<attr name=\"nutritionalValue\">?</attr>"}
17
18      {?CreateXMLTagWithContent
19          {?ValueGetLocal {&Nutritional programs.Nutritional program}},
20          "<attrGroupMany name=\"nutritionalProgramIngredients\">?</attrGroupMany> "}
21  </row>
22  }{!Reset variable}{?ValueSetLocal {&Nutritional programs.Nutritional program},""}

```

Export template module: *Health related information*

Changes: lines 25-32 replace former lines 25-43

Health related information

```

1      {?CreateXMLTagWithContent {?EnumerationKey {&Health related information.Is not intended for consumption}}
2      , "<attr name=\"isProductChemicalNotIntendedForHumanConsumption\">?</attr>"}
3
4      {?CreateXMLTagWithContent {?SplitKeywords {?EnumerationKey {&Health related information.Nutrition label type code}}
5      , "<value>", "</value>"}
6      , "<attrMany name=\"nutritionLabelTypeCode\">?</attrMany>"}
7
8      {?CreateXMLTagWithValue {?EnumerationKey {&Health related information.Cannabis CBD type}}
9      , "<attr name=\"cannabisCBDTypeCode\">?</attr>"}
10
11     {?CreateXMLTagWithContent {?ValueGetLocal "tmpNutritionalProgram"}, "<attrGroupMany name=\"nutritionalProgram\">?</
    attrGroupMany>"}

```


Export template module: *Items*

Changes: lines 140-141 have been added

Items	
1	<code>{ \$Collect trade item handling stacking }</code>
2	<code>{ \$Collect nutritional program ingredients }</code>
3	<code>{ \$Collect nutritional programs }</code>
4	<code>{ \$Collect fish meat poultry content }</code>

Nutrient basis quantity type: By Package or Container

Export template module: *Collect nutrients*

Changes: lines 67-96 have been added

Collect nutrients	
1	<code>{ !CollectByContainer }</code>
2	<code>{ ?ValueSetLocal "tmpNutrientListByContainerName", ByContainerName_{&Nutrients.Preparation state}_{&Nutrients.Target market} }</code>
3	<code>{ ?ValueSetLocal { ?ValueGetLocal "tmpNutrientListByContainerName", { ?ValueGetLocal { ?ValueGetLocal "tmpNutrientListByContainerName" } }</code>
4	
5	<code>{ ?IfNotEmptyThenNotEnc {&Nutrients.Percentage of daily intake (By Package or Container, UOM type)}{&Nutrients.Quantity contained (By Package or Container, UOM type)}{&Nutrients.Quantity contained (By Package or Container, UOM type second)}{&Nutrients.Measurement precision (By Package or Container, UOM type)}{&Nutrients.Measurement precision (By Package or Container, UOM type second)},</code>

```

6 <row>
7   {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Nutrient type}} ,<attr name="\nutrientTypeCode\">?</attr>}
8
9   {?CreateXMLTagWithValue {?FormatDecimal {&Nutrients.Percentage of daily intake (By Package or Container, UOM type)},"",0
   {?IfEmptyThen {&Nutrients.Percentage of daily intake (By Package or Container, UOM type)} ,{?FormatDecimal
   {&Nutrients.Percentage of daily intake (By Package or Container, UOM type second)},"",0}}
10     ,<attr name="dailyValueIntakePercent">?</attr>"
11   }
12
13   {?CreateXMLTagWithValue  {?EnumerationKey {&Nutrients.Measurement precision (By Package or Container, UOM type)}} {?
   IfEmptyThen {&Nutrients.Measurement precision (By Package or Container, UOM type)} ,{?EnumerationKey
   {&Nutrients.Measurement precision (By Package or Container, UOM type second)}}}
14     ,<attr name="measurementPrecisionCode">?</attr>"
15   }
16
17   {?CreateXMLTagWithValue {?EnumerationKey {&Nutrients.Expressed as part of (By Package or Container, UOM type)}}{?
   IfEmptyThen {&Nutrients.Expressed as part of (By Package or Container, UOM type)} ,{?EnumerationKey {&Nutrients.Expressed
   as part of (By Package or Container, UOM type second)}}}
18     ,<attr name="expressedAsPartOf">?</attr>"
19   }
20
21   {?IfNotEmptyThenNotEnc {&Nutrients.Quantity contained (By Package or Container, UOM type)}{&Nutrients.Quantity
   contained (By Package or Container, UOM type second)},
22     "<attrQualMany name="quantityContained">
23     {?CreateXMLTagWithValue {?FormatDecimal {&Nutrients.Quantity contained (By Package or Container, UOM type)},".", "6"}
24   ,
25     "<value qual="{&Nutrients.Quantity contained UOM (By Package or Container, UOM type).Code (Unit system)}">?</
   value>"}
26     {?CreateXMLTagWithValue {?Compare {%UOM type},{%UOM type second}},,{?FormatDecimal {&Nutrients.Quantity contained
   (By Package or Container, UOM type second)},".", "6"}},
27     "<value qual="{&Nutrients.Quantity contained UOM (By Package or Container, UOM type second).Code (Unit system)}">?
   </value>"}

```

```

27     </attrQualMany>"
28   }
29 </row>
30 }}

```

Export template module: *Nutritional information*

Changes: lines 1-30 have been added

Nutritional information

```

1  {!BY_PACKAGE_OR_CONTAINER only export when there is data for "by container"}
2  {?ValueSetLocal "tmpNutrientListByContainerName",ByContainerName_{{&Nutritional information.Preparation state}}
3  _{{&Nutritional information.Target market}}}
4  {?IfNotEmptyThenNotEnc {?ValueGetLocal {?ValueGetLocal "tmpNutrientListByContainerName"}},
5    <row>
6    <attr name ="preparationStateCode">{{EnumerationKey {{&Nutritional information.Preparation state}}}}</attr>
7
8    {?IfNotEmptyThenNotEnc {{&Nutritional information.Daily value intake reference (Language)}}{{&Nutritional
9    information.Daily value intake reference (Language second)}},
10    "<attrQualMany name="dailyValueIntakeReference">
11    {?CreateXMLTagWithValue {{&Nutritional information.Daily value intake reference (Language)}},
12    "<value qual="GDSNEnumerationCode {{&Nutritional information.Language (Language)}},GDSN"}>?</value>"}
13    {?IfNotEmptyThenNotEnc {?ValueGet "LanguageDiffers"}, {?CreateXMLTagWithValue {{&Nutritional information.Daily value
14    intake reference (Language second)}},
15    "<value qual="GDSNEnumerationCode {{&Nutritional information.Language (Language second)}},GDSN"}>?</value>"}
16    </attrQualMany>"
17  }

```

```

16     <attr name="nutrientBasisQuantityTypeCode">BY_PACKAGE_OR_CONTAINER</attr>
17
18     {?CreateXMLTagWithValue {?FormatDecimal {&Nutritional information.Nutrient basis quantity (UOM type)},.,6,},
19     "<attrQual name="nutrientBasisQuantity" qual="{&Nutritional information.Nutrient basis quantity UOM (UOM type).Code
    (Unit system)}">?</attrQual>"
20     }
21     {!IM accepts here multi values which we currently do not support }
22     {?CreateXMLTagWithContent {?FormatDecimal {&Nutritional information.Serving size (UOM type)},.,6,},
23     <attrQualMany name="servingSize"><value qual="{&Nutritional information.Serving size UOM (UOM type).Code (Unit
    system)}">?</value></attrQualMany>
24     }
25
26     {?CreateXMLTagWithContent {?ValueGetLocal {?ValueGetLocal "tmpNutrientListByContainerName"}}, "<attrGroupMany
    name=\"nutrientDetail\">?</attrGroupMany>"}
27     {!Reset variable}{?ValueSetLocal {?ValueGetLocal "tmpNutrientListByContainerName"}, ""}
28     </row>
29 }
30
31 {!BY_MEASURE only export when there is data for "by measure"}

```

2.1.18.5 Compatibility Export Template for older versions

The relevant changes of GDSN Major Release 3.1.15 will be supported from version 10.0.0.04 upwards. All older versions won't have any of the changes mentioned in this migration guide so we highly recommend to update to the latest version. For all other customers we will provide a compatibility export template.

This template makes it still possible to send the data to the pool but does not contain any of the new fields or adjustments of existing fields. To make the export template backwards compatible we did the following changes:

CR_CatalogueRequest Item.ext (IM)

Export template module: *Ingredient information*

Changes: lines 3-10 have been changed

Ingredient information	
1	{?CreateXMLTagWithContent
2	{?CreateXMLTagWithValue {&Ingredient information.Ingredient statement (Language)},
3	"<value qual=\"{?GDSNEnumerationCode {&Ingredient information.Language (Language)}, "GDSN"}\">?</value>"}
4	{?IfNotEmptyThenNotEnc {?ValueGet "LanguageDiffers"},
5	{?CreateXMLTagWithValue {&Ingredient information.Ingredient statement (Language second)},
6	"<value qual=\"{?GDSNEnumerationCode {&Ingredient information.Language (Language second)}, "GDSN"}\">?</value>"}
7	}
8	, "<attrGroupMany name=\"ingredientStatement\"><row><attrQualMany name=\"statement\">?</attrQualMany></row></attrGroupMany>"}

Export template module: *Health related information*

Changes: lines 25-53 have been changed

Health related information	
1	{?CreateXMLTagWithContent {?SplitKeywords {?EnumerationKey {&Health related information.Nutrition label type code}}}
2	,<value>, "</value>"}
3	,<attrMany name=\"nutritionLabelTypeCode\">?</attrMany>"}

```

4      {?CreateXMLTagWithValue {?EnumerationKey {&Health related information.Cannabis CBD type}}
5      , "<attr name=\"cannabisCBDTypeCode\">?</attr>"}
6
7      {?CreateXMLTagWithContent
8      {?SplitKeywords {?EnumerationKey {&Health related information.Nutritional program code}}
9      {!before each program code}, "<row><attr name=\"nutritionalProgramCode\">"
10     {!after each program code}, "</attr>"
11     {?CreateXMLTagWithValue {&Health related information.Nutritional score}
12     , "<attr name=\"nutritionalScore\">?</attr>"}
13     {?CreateXMLTagWithValue {?FormatDecimal {&Health related information.Nutritional value}, ".", 6}
14     , "<attr name=\"nutritionalValue\">?</attr>"}
15     {?CreateXMLTagWithContent
16     {?IfNotEmptyThenNotEnc {&Health related information.Nutritional program ingredient value (UOM type)}
17     , "<attrQualMany name=\"nutritionalProgramIngredientMeasurement\">"
18     {?CreateXMLTagWithValue {?FormatDecimal {&Health related information.Nutritional program ingredient value (UOM
19     type)}, ".", "6", }
20     , "<value qual=\"{&Health related information.Nutritional program ingredient unit (UOM type).Code (Unit
21     system)}\">?</value>"}
22     </attrQualMany>"
23     }
24     {?CreateXMLTagWithValue {?EnumerationKey {&Health related information.Nutritional program ingredient type}}
25     , "<attr name=\"nutritionalProgramIngredientTypeCode\">?</attr>"}
26     , "<attrGroupMany name=\"nutritionalProgramIngredients\"><row>?</row></attrGroupMany>"
27     }
28     </row>
29     }
30     , "<attrGroupMany name=\"nutritionalProgram\">?</attrGroupMany>"
31     }

```

CIN_CatalogItemNotification.ext (DSE/Atrify)

There are no changes necessary. The old template should still be compatible to the new major version

2.1.18.6 B2B

With the Update to GDSN 3.1.15 the XSDs of DSE/Atrify and IM changed. These are used by the data transformations in B2B.

If you are a **cloud customer**, you don't need to do anything. A new EC2 instance with an updated B2B will be created for you from a new AMI.

If you use an **on premise** installation of B2B, you need to execute the following steps to update your system:

1. In your Power Center installation find the folder `DataTransformation/ServiceDB` and make a backup of this folder.
2. Stop the GDSN workflows.
3. Stop Power Center, Data Exchange and MFT.
4. In the `PIM_<version>_resources_gdsn.delta.zip` of the GDSN Accelerator Package in the folder `Common` you will find a file called `B2B_GDSN.zip`.
5. Extract all files and open `DT_Services` folder.
6. Depending on whether you use GDSN as a
 - a. Data Source (e. g. manufacturer), copy the content of the `DataSource` folder to the `DataTransformation/ServiceDB` folder of your B2B installation.
 - b. Data Recipient (e. g. retailer), copy the content of the `DataRecipient` folder to the `Data Transformations/ServiceDB` folder of your B2B installation.
7. Start Power Center, Data Exchange and MFT.
8. Start the workflows.

2.2 GDSN Accelerator with OpenAS2

2.2.1 Why using OpenAS2?

For companies interested in sharing information with the GDSN only, and not leveraging the extensive trade partner management capabilities of Informatica B2B, using the OpenAS2 software is a much more light-weight solution which especially in cost-sensitive cloud scenarios can provide a big advantage. As OpenAS2 is a very small application focused on the single use case to send/receive files using the AS2 protocol:

- no database server is required
- less memory is required (~2 GB for files up to 300 MB)
- less CPU power is required

Using fewer resources minimizes the costs of the corresponding infrastructure where the application runs. Due to the focusing on this specific use case the installation and configuration efforts are lower, and startup times are quicker, which in return minimizes downtimes and the provisioning of the system.

2.2.1.1 Having everything in one place

To support the GDSN choreography using OpenAS2, many functionalities have been introduced as standard Product 360 capabilities, leveraging for example the hot folder or the import. The only part which is left to OpenAS2 is encrypting / decryption the messages and the AS2 connection to the GDSN data pool. Having all the functionality in one place is an enormous advantage for any GDSN update which is made on a regular basis. You only have to update one application which lowers the efforts and costs, as well as minimizes the risk any update might contain.

Also when it comes up to customization and adjustment of standard functionality, it is easier for our customers and partners, as they are usually already familiar with the processes, capabilities and the technology of Product 360.

Finally, it is easier for any user as most of the steps including errors are now shown in Product 360 directly and it's not required anymore to check multiple applications to trace the complete process.

2.2.2 Supported Scenarios

Currently only the **1WorldSync Item Management - Data Source** scenario is supported. Neither the atrify (DSE) or other GDSN scenario is supported, nor any Data Recipient scenario. For both, this solution is generally possible but will need additional implementation efforts which are not supported out of the box right now.

2.2.3 Official OpenAS2 documentation

In the installation package of OpenAS2 you can find the OpenAS2HowTo.pdf file which contains a lot of useful information about OpenAS2 in general but also about installation and configuration possibilities.

Although most of things are described in the chapters below, it can help to find solutions in case of any error or if more details are necessary to any configuration.

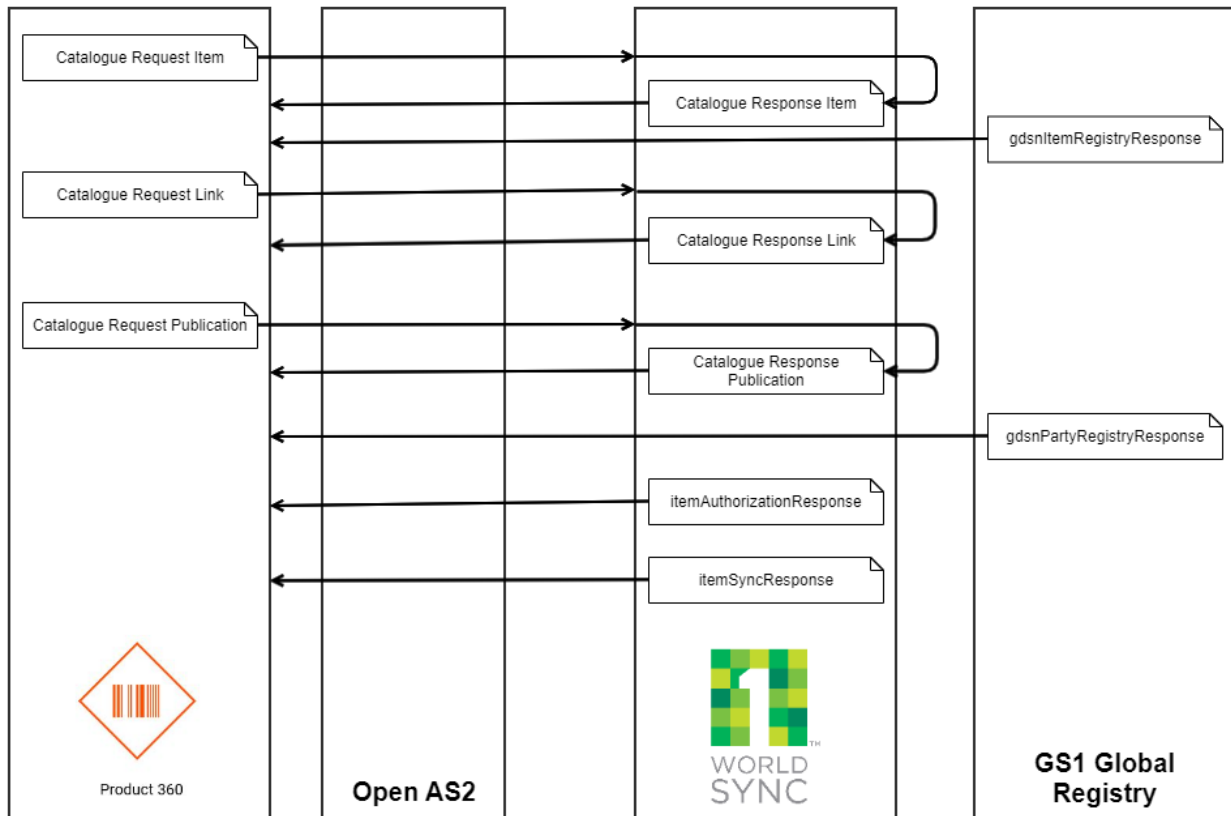
2.2.4 FieldList

As we only changed the "Middleware", nothing within Product 360 related to the data model changed. So the supported field list is the same as before and can be found in the GDSN Accelerator (Major Release 3) documentation.

2.2.5 Choreography

2.2.5.1 IM message choreography

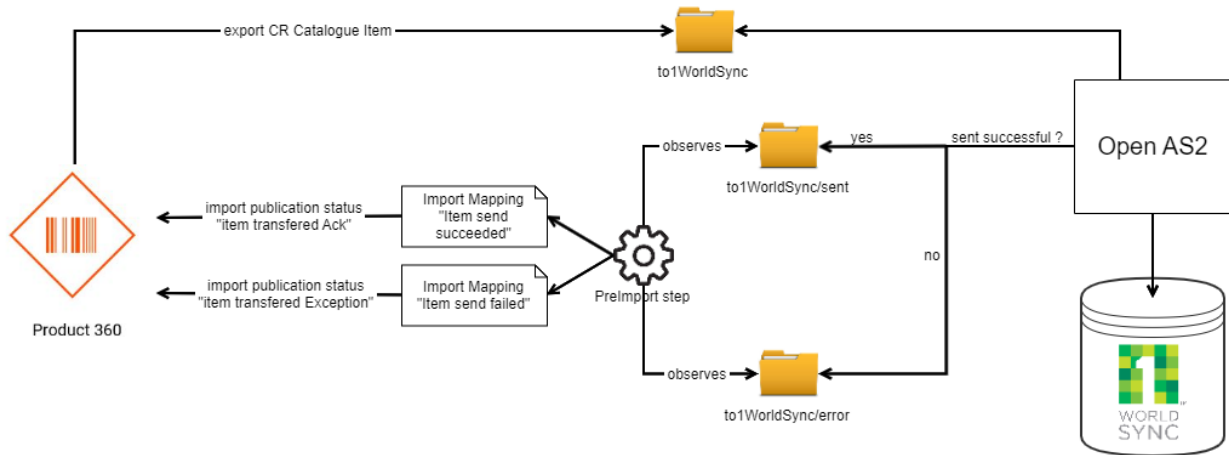
By sending a "Catalogue Request" message of type "Item" new items are added or existing ones are modified in the GDSN pool. After all necessary items have been successfully added to the GDSN pool, hierarchies can be created by linking these items together. Items can also be removed from hierarchies by unlinking them again. This is done by sending a "Catalogue Request" message of type "Link". When all needed item data has been added to the GDSN pool and the hierarchies were created, it is possible to publish it for a specific customer (data recipient) or market groups by sending a "Catalogue Request" of type "Publication" message. By doing this the data recipient has the ability to view and synchronize the published item including all the child items below that item. An existing publication for a specific customer or market group can be deleted.



2.2.5.2 The choreography "send Catalogue Request item" in more detail

With OpenAS2 not much things changed. Product 360 still leverages the export and put the GDSN message file to a folder. Instead of Informatica B2B, OpenAS2 will pick it up, encrypt it and send it to 1WorldSync. After receiving the confirmation, that the message could be successful delivered*, the message will be put to sent folder. In case something goes wrong during the encryption or sending process, OpenAS2 will try to resend it several times and if it was still not successful the message will be put to the error folder.

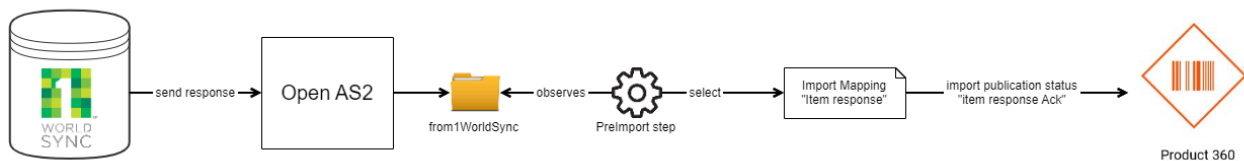
The sent as well as the error folder are configured as hot folders in Product 360 and so files will be picked up and processed by a preparatory import step. This preparatory import step will analyze the name and/or the content of the file and select the corresponding mapping to import a Publication status for each item in the file. Creating this Publication status was done before by Informatica B2B via the Service API of Product 360.



i *Please note that "delivered" does only mean that 1WorldSync received the message, but neither opened nor validated it. The real Catalogue Response will be sent asynchronously and is shown in the next chapter.

2.2.5.3 The choreography "receive Catalogue Response item" in more detail

After sending out a Catalogue Request item, 1WorldSync will send a Catalogue Response item asynchronously. OpenAS2 will receive the response, decrypt it and put it to the folder "from1WorldSync" which is also observed by Product 360. Also here the configured preparatory import step will analyze the name and content of the file and selects the corresponding mapping to import a publication status for each item in the file.




2.2.5.4 Other messages

All other messages as the "Catalogue Request Link", "Catalogue Request publication" etc... will be processed the same way as the "Catalogue Request Item" which was visualized above.

2.2.6 Installation / Configuration of OpenAS2

2.2.6.1 Install OpenAS2

1. Go to <https://sourceforge.net/projects/openas2/files/latest/download> and download the OpenAS2Server zip file.
2. Unzip the file to a directory of your choice, e.g. "C:\OpenAS2". This directory will be referenced in this documentation as <OpenAS2>


 The path to your OpenAS2 directory should not contain any whitespaces. This could lead to errors when starting OpenAS2.

2.2.6.2 Configure OpenAS2

There are two main configuration files which are relevant to each installation and have to be configured individually for every customer:

config.xml

This file contains common configurations. It configures the modules that will be activated by OpenAS2 server when it starts up. You can use the default template provided by OpenAS2 or you download and use our config.xml which has the properties and the modules pre-configured. Replace the {customer_as2_url} placeholder with your customer's URL and try it out.

 The following chapter is only important if you change the OpenAS2 file structure. If you go with the defaults provided by Informatica nothing has to be adjusted or considered.

Sending Files

OpenAS2 has a directory polling module that scans configured directories for files and will send the file to a partner (e.g. 1WorldSync). Some of the key attributes for the polling module defined in the config.xml are:

- *outboxdir*: specifies the directory to scan for files to send

```
outboxdir="$properties.storageBaseDir$/to1WorldSync"
```

- *errordir*: specifies the directory where the file will be put to whenever OpenAS2 is not able to send it due to any configuration or connection issue

```
errordir="$properties.storageBaseDir$/to1WorldSync/error"
```

- *stored_error_filename*: defines the name used to store the file in the errordir

```
stored_error_filename="OPENAS2-
$rand.UUID$@$msg.attributes.filename$_failed_to1WorldSync"
```

- *sentdir*: specifies the directory where the files that are successfully sent will be stored

```
sentdir="$properties.storageBaseDir$/to1WorldSync/sent"
```

- *stored_sent_filename*: defines the name used to store the file in the sentdir that are successfully sent

```
stored_sent_filename="OPENAS2-
$rand.UUID$@$msg.attributes.filename$_succeeded_to1WorldSync"
```

Receiving Files

There is not much to configure for receiving files. Messages will be received and the files stored in the configured directory. The default modules for receiving and string files is shown below.

```
<module classname="org.openas2.processor.storage.MessageFileModule"
    filename="$properties.storageBaseDir$/from1WorldSync/inbox/$msg.content-
disposition.filename$-$msg.headers.message-id$_from1WorldSync"
    header="$properties.storageBaseDir$/from1WorldSync/msgheaders/$date.yyyy-
MM-dd$/msg.content-disposition.filename$-$msg.headers.message-id$"
    protocol="as2"
    tempdir="$properties.storageBaseDir$/temp"/>
```

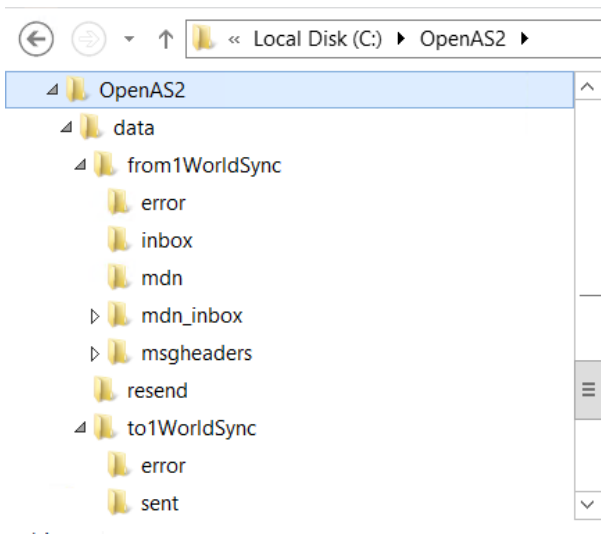
```
<module classname="org.openas2.processor.receiver.AS2ReceiverModule"
    port="5080"
    errordir="$properties.storageBaseDir$/from1WorldSync/inbox/error"
    errorformat="sender.as2_id, receiver.as2_id, headers.message-id"/>
```

partnerships.xml

The partnerships.xml contains information about the connection details between the trade partners such as AS2 ids, certificates and URLs. You can use the default template provided by OpenAS2 or you download and use our partnerships.xml which is pre-configured for a connection to the PreProd 1WorldSync data pool. Replace the {customer} placeholders with your customer's name, email etc... and try it out.

Overview:

If the default configurations are used your file structure should look like in the screenshot below:



2.2.6.3 Certificates

Finally you have to import and/or create certificates to communicate secure with 1WorldSync. This is described in the next chapter.

2.2.6.4 Troubleshooting

Disposition exception processing MDN

If you get the following exception when you try to send a file via OpenAS2 you should check the AS2 ids you have configured in partner.xml. This error is usually caused by an incorrect configuration. You have to use the AS2 ids agreed with the partner.

Disposition exception processing MDN

```
2020-05-29 16:19:57.856 ERROR AS2Util: Disposition exception processing MDN ...
[<OPENAS2-29052020161957+0000-ac9611f6-9686-4b18-a949-
fe7e517502ab@openas2_dev_as2_id_0838016003001>]
org.openas2.DispositionException: null/null; null
    at org.openas2.util.DispositionType.validate(DispositionType.java:98)
    at org.openas2.util.AS2Util.checkMDN(AS2Util.java:183)
    at org.openas2.util.AS2Util.processMDN(AS2Util.java:509)
    at
org.openas2.processor.sender.AS2SenderModule.processResponse(AS2SenderModule.java:234
)
    at
org.openas2.processor.sender.AS2SenderModule.sendMessage(AS2SenderModule.java:207)
    at
org.openas2.processor.sender.AS2SenderModule.handle(AS2SenderModule.java:128)
    at org.openas2.processor.DefaultProcessor.handle(DefaultProcessor.java:65)
```

```

        at
org.openas2.processor.receiver.MessageBuilderModule.processDocument(MessageBuilderMod
ule.java:192)
        at
org.openas2.processor.receiver.DirectoryPollingModule.processFile(DirectoryPollingMod
ule.java:191)
        at
org.openas2.processor.receiver.DirectoryPollingModule.updateTracking(DirectoryPolling
Module.java:165)
        at
org.openas2.processor.receiver.DirectoryPollingModule.poll(DirectoryPollingModule.jav
a:76)
        at
org.openas2.processor.receiver.PollingModule$PollTask.run(PollingModule.java:62)
        at java.util.TimerThread.mainLoop(Unknown Source)
        at java.util.TimerThread.run(Unknown Source)

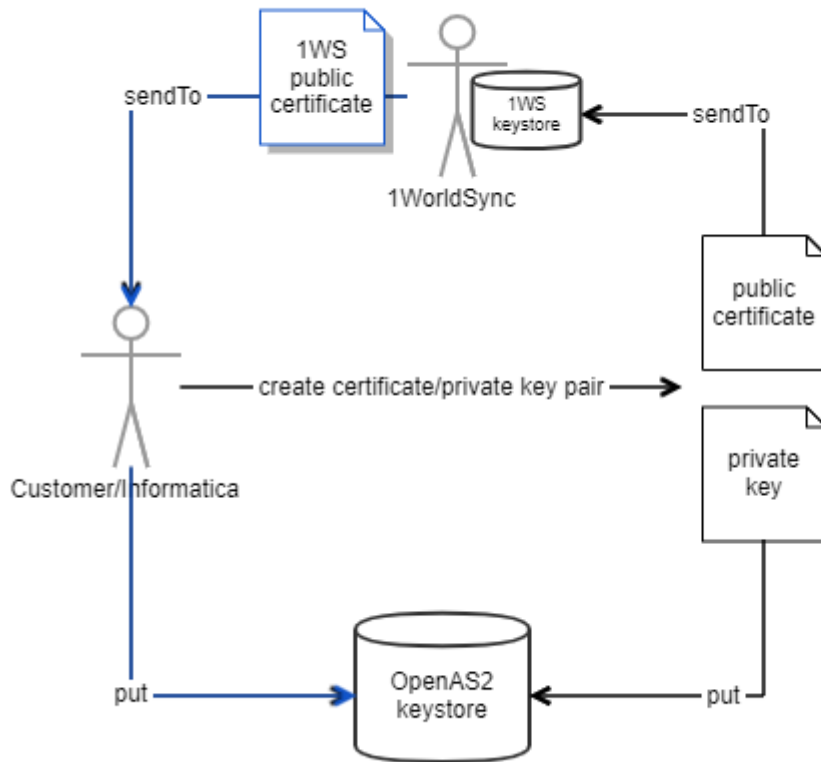
```

2.2.6.5 Certificates

Overview

To have a successfully and secured AS2 connection, certificates have to be exchanged between both trade partners. For on premise installations the customer has to do this, for hosted installation Informatica has to do this part. As the diagram below shows, the customer/Informatica has to create a public certificate / private key pair, send the public certificate to 1WorldSync (via EMail) and put the private key to the key store of OpenAS2. Then the public certificate of 1WorldSync should be retrieved (by EMail or download from the webpage) and also put to the OpenAS2 key store.

Certification exchange



Create a private key / public certificate

OpenAS2 provides a command to create a private key / public certificate pair. The public certificate has to be sent to 1WorldSync so they can encrypt their messages with this certificate. To create the pair go to the <OpenAS2>/config directory and execute (you can find the password in the config.xml file)

```

..\bin\gen_p12_key_par.bat <keyStore> <Customer>_<env>_to_1ws <sigAlg>
"CN=<CommonName>, O=<Organization>, OU=<OrganizationalUnit>, L=<Locality>, S=<State>,
C=<Country>"

```

or in a concrete example for Informatica

```

..\bin\gen_p12_key_par.bat as2_certs customer_dev_to_1ws sha256 "CN=informatica.com,
O=Informatica GmbH, OU=R&D, L=Stuttgart, S=BW, C=GER"

```

The result should look like:


```

PS C:\Informatica\AS2\testcertificate> ..\bin\gen_p12_key_par.bat as2_certs informatica_to_1ws sha256 "CN=informatica.com, O
=Informatica GmbH, OU=R&D, L=Stuttgart, S=BW, C=GER"
DNAM = "CN=informatica.com, O=Informatica GmbH, OU=R&D, L=Stuttgart, S=BW, C=GER"
Generate a certificate to a PKCS12 key store.
Generating certificate: using alias informatica_to_1ws to as2_certs.p12"
Enter password for keystore:testas2
Certificate stored in file <informatica_to_1ws.cer>

Generated files:
PKCS12 keystore: as2_certs.p12
Public Key File: informatica_to_1ws.cer

PS C:\Informatica\AS2\testcertificate> |

```

You now can send the public certificate (in this case informatica_to_1ws.cer) to 1WorldSync and either use the generated key store or update an existing one.

How to import a new certificate

1. Open a console and navigate to the <OpenAS2>/config
2. Execute following command:

```

..\bin\import_public_cert.bat c:\Informatica\OpenAS2\config\<yourCertificate>
<keyStore> <alias> <action>

```

or in example:

```

..\bin\import_public_cert.bat c:
\Informatica\OpenAS2\config\as2_preprod_1worldsync_com_10Jan2020_der.cer c:
\Informatica\OpenAS2\config\as2_certs.p12 from_1ws replace

```

replace is optional and only needed if the certificate already exists.

3. The password for the keystore can be found in the config.xml of OpenAS2 at the XML element <certifications>. On suv12rddemo07 it is "testas2", default is "changeit"

How to list all certificates (check)

1. To list all certificates of your key store

```

keytool -list -keystore <keystore> -storepass <password> -storetype PKCS12

```

or in example:

```

keytool -list -keystore as2_certs.p12 -storepass testas2 -storetype PKCS12

```

2. Your keystore should look like this:

```
# keytool -list -keystore as2_certs.p12 -storepass testas2 -storetype PKCS12
Keystore type: PKCS12
Keystore provider: SunJSSE

Your keystore contains 4 entries

partnera, Sep 6, 2018, PrivateKeyEntry,
Certificate fingerprint (SHA1): 2D:4B:42:05:56:80:9B:5D:0E:63:4D:4A:23:3D:9A:39:C3:8D:51:21
mycompany, Sep 6, 2018, PrivateKeyEntry,
Certificate fingerprint (SHA1): 1E:16:65:9B:7A:F2:59:EA:B7:B7:4F:E5:EB:D3:CF:89:3A:0F:89:CA
infa_to_1ws, Mar 30, 2020, PrivateKeyEntry,
Certificate fingerprint (SHA1): C2:80:BD:CE:81:22:EE:FF:26:0B:41:FC:04:AB:94:A7:CB:50:3D:50
1ws, Mar 30, 2020, trustedCertEntry,
Certificate fingerprint (SHA1): B1:5C:18:25:05:08:BD:80:33:44:B3:04:9B:D5:58:3D:7F:7A:DD:91
```

How to delete a certificate

To delete a certificate of your key store

```
keytool -delete -keystore <keystore> -storepass <password> -alias <alias>
```

or in example:

```
keytool -delete -keystore as2_certs.p12 -storepass testas2 -alias 1ws
```

2.2.7 Configuration of Product360

2.2.7.1 Import Hotfolder configuration

Hotfolder configurations

The communication between Product360 and OpenAS2 is completely file based. Whenever OpenAS2 receives a message file (e.g. a Publication response), the file will be decrypted and then put to the configured folder. This folder has to be observed by Product360 to pick up the responses and import the corresponding publication status. This chapter describes how to configure the hotfolders correctly in order to make the choreography work.

System configuration

1. Navigate to the configuration folder of your Product360 server and open the file server.properties.
2. Find and set the property inbox.hotfolders = \${filestorage.dir.shared}/inbox/hotfolder;<OpenAS2>/data/from1WorldSync/inbox;<OpenAS2>/data/to1WorldSync/error;<OpenAS2>/data/to1WorldSync/sent . Please be sure to have no spaces between the configurations as this leads to an error during startup.

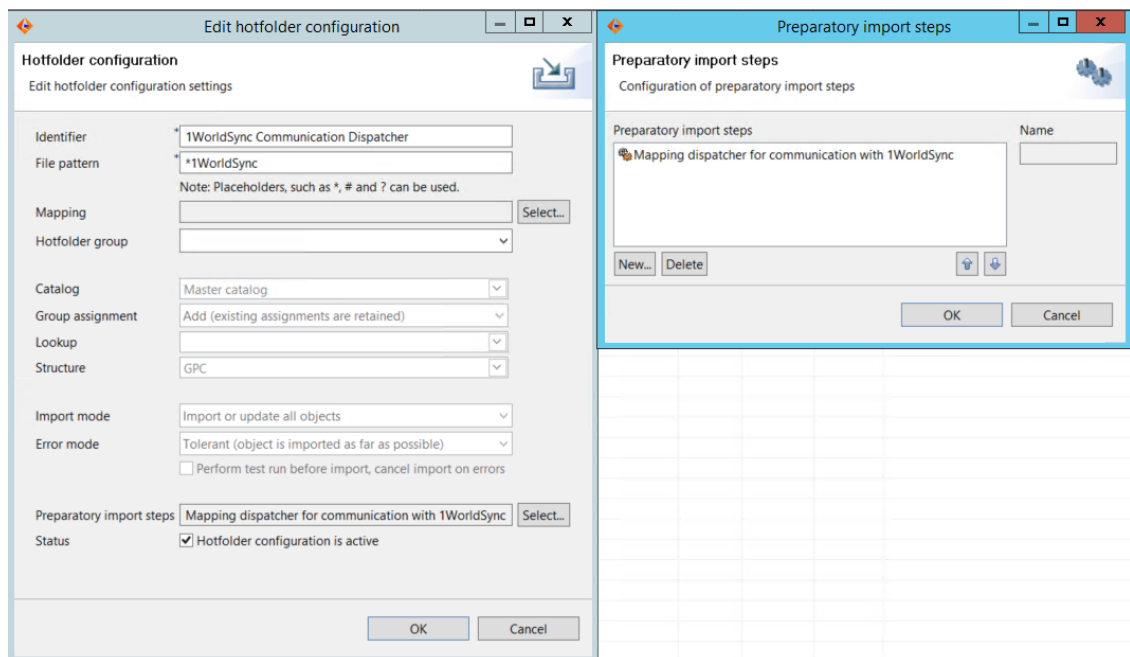
Hotfolder configuration

1. The hotfolder configuration needs import mappings which are provided by Informatica. The import mappings are located in the GDSN accelerator package under DataSource/IM/OpenAS2/Import Mappings. Import the mappings to Product360 by following the steps below:
 - Go to Import perspective in the desktop client
 - Click on *New Import project*
 - In the dialog select *Load existing mapping from file* and select your file. Click *finish* afterwards
 - Click *Save as* and save the mapping. Set the *Category* to GDSN and please be sure that the *Purpose* is set to Hotfolder

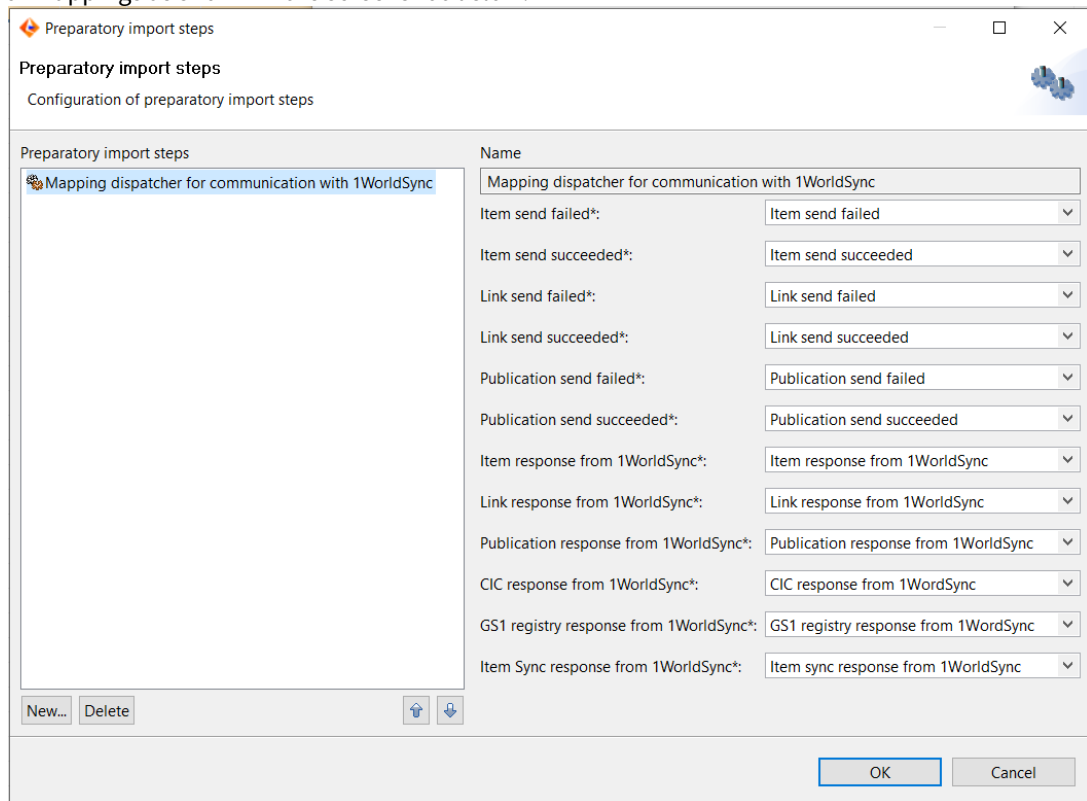
We provide a pre-import step which dispatches the imported mappings to ensure the communication with 1WorldSync.

2. Go to the Hotfolder perspective and create a hotfolder configuration with the following settings:

- Identifier: 1WorldSync Communication Dispatcher
- File pattern: *1WorldSync
- Preparatory import steps : Mapping dispatcher for communication with 1WorldSync. See the screenshot below.

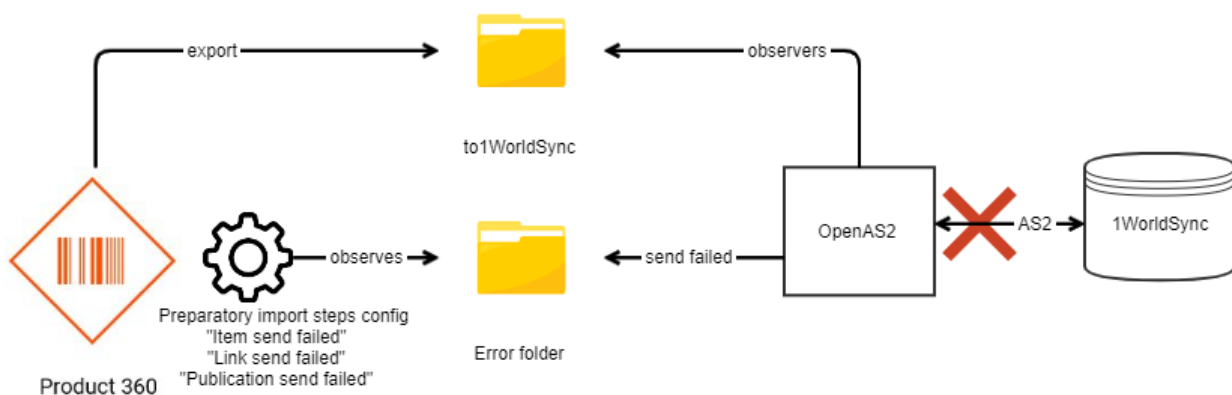


- In the Preparatory import steps dialog, select the right mapping for each communication from the list of mappings as shown in the screenshot below:



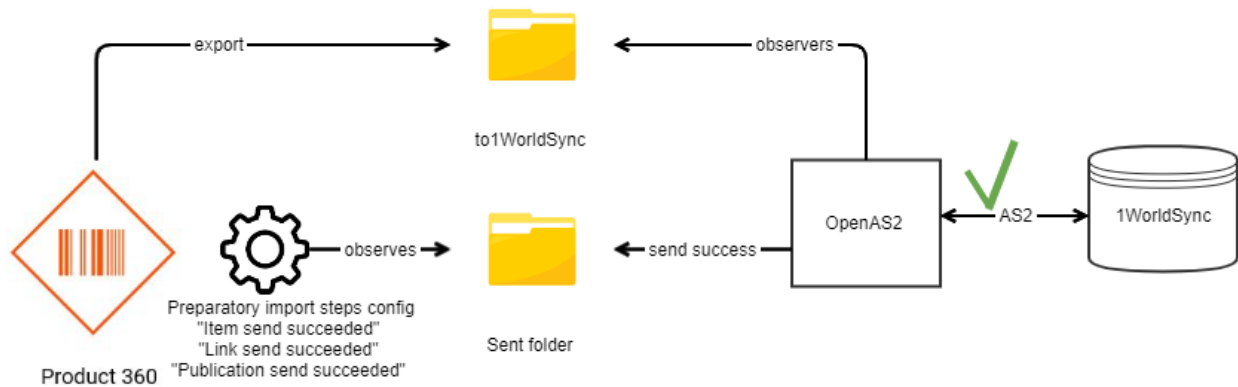
"Item/Link/Publication send failed" communication

Whenever OpenAS2 is not able to send a file due to any configuration or connection issue, the file will be put to the configured error folder (see the attribute *error_dir* in the corresponding module of your config.xml). We have to configure the pre-import step to import this file and mark all corresponding items/links/publications as not sent by setting the publication status to "Item/Link/Publication transferred -> Exception".



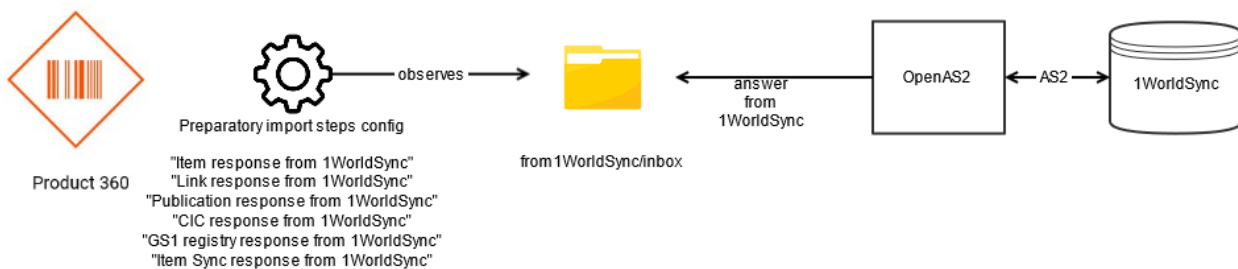
"Item/Link/Publication send succeeded" communication

Whenever OpenAS2 succeeds to send a file, the file will be archived and put to the configured sent folder (see the attribute *sentdir* in the corresponding module of your config.xml). We have to configure the pre-import step to import this file and mark all corresponding items/links/publications as sent successfully by setting the publication status to "Item/Link/Publication transferred -> Acknowledgement".



"Import responses" communication

We provide the same pre-import step which analyses the message files from 1WorldSync and uses the right import mapping and import the corresponding publication status for an item, link, publication, CIC, GS1 registry or item sync response.

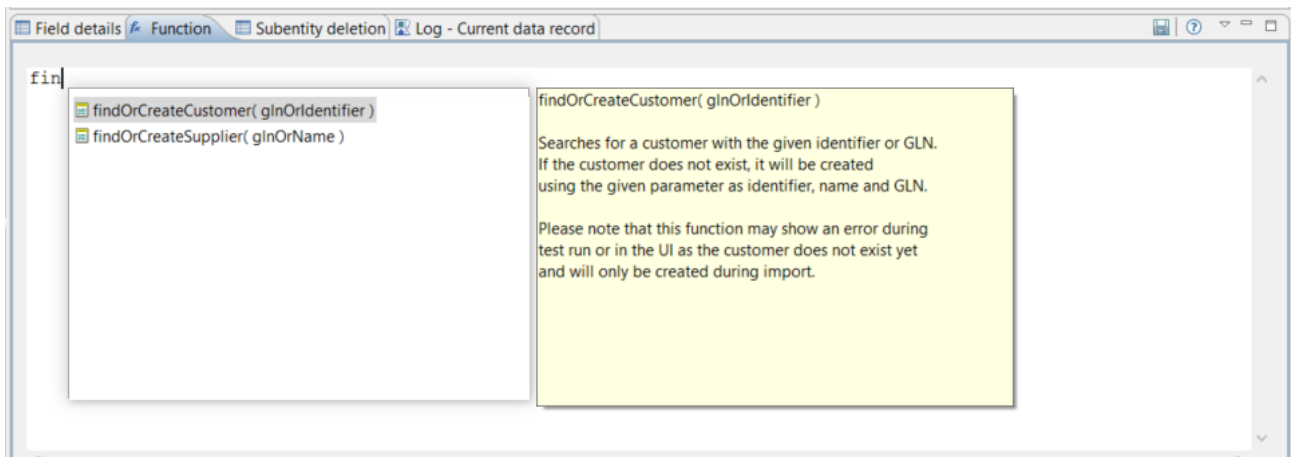


Import functions to create customers and suppliers

There may be responses containing unknown customers and suppliers. This is especially true for the CatalogueItemConfirmation, that contains feedback to your items from unknown parties using the Recipient Scenario.

In order to create them in Product360, there are two new import functions "findOrCreateCustomer" and "findOrCreateSupplier". These are used by our out-of-the-box import mappings provided for the hotfolder configuration.

First they look if a customer/supplier already exists in the system. If multiple results are found, the first one is taken. If no result is found, the customer/supplier is created in the system with the given GLN as identifier, name and GLN.



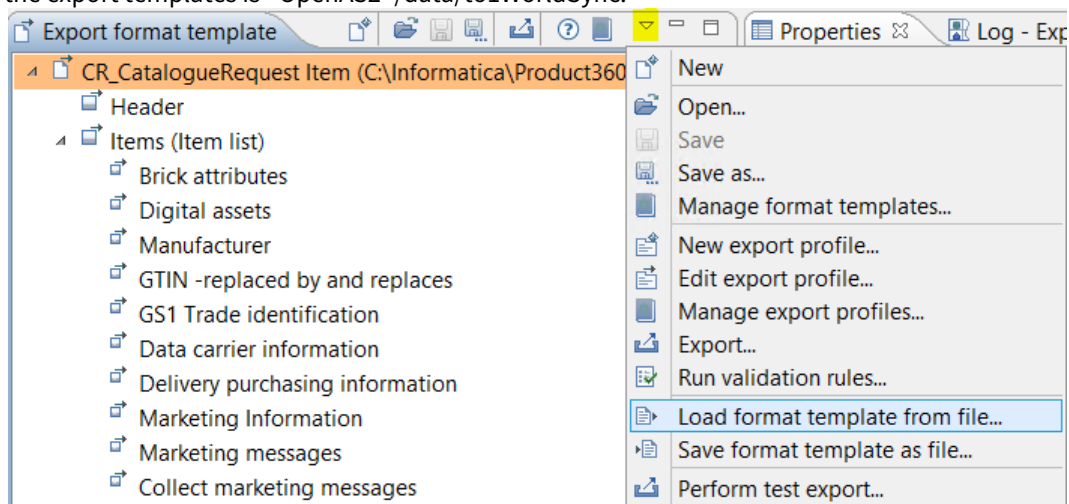
See the tooltips for further information.

2.2.7.2 Export

Load export templates

In order to send out messages to the GDSN data pool, the export templates has to be loaded to Product360 by following the steps below:

- Go to Export format templates perspective in the desktop client.
 - In the View Menu select *Load format templates from file* and select your file. See the screenshot bellow. You can find the export templates in the GDSN accelerator package under DataSource/IM/OpenAS2/ExportTemplates/CatalogueRequest.
- Please note that you may have to adjust the folder the messages get exported to. The default folder set in the export templates is <OpenAS2>/data/to1WorldSync.



- Set the *Category* to GDSN and the *Purpose* will be automatically set to Immediate export.
- Click *Save export format template* and save the export template.

Overview:

Finally your export format templates should look like in the screenshot below:

Export templates name



The following chapter is only important if you change the file names of the GDSN messages. If you go with the defaults provided by Informatica nothing has to be adjusted or considered.

The name of the GDSN messages is essential for the whole OpenAS2 process. Based on GDSN message name, the hotfolder preparatory import step will provide the right import mapping to import the corresponding publication status. Make sure you configure your export templates to use a name following the pattern as for example `.*CR_CatalogueRequest Item.*` for Item send succeeded/failed communication (*CR_CatalogueRequest Item* is a fixed part and it can be preceded or followed by any other string). By default the GDSN message files will have the same name as the export template and so everything will work without any adjustments. Below are the different file patterns that should be respected to insure a working flow.

Item send pattern : `.*CR_CatalogueRequest Item.*_failed_to1WorldSync`
 Example: `myPrefix_CR_CatalogueRequest Item_myPostfix_failed_to1WorldSync`

Item send pattern: `.*CR_CatalogueRequest Item.*_succeeded_to1WorldSync`
 Example: `myPrefix_CR_CatalogueRequest Item_myPostfix_succeeded_to1WorldSync`

Link send pattern: `.*CR_CatalogueRequest Link.*_failed_to1WorldSync`
Example: `myPrefix_CR_CatalogueRequest Link_myPostfix_failed_to1WorldSync`

Link send pattern: `.*CR_CatalogueRequest Link.*_succeeded_to1WorldSync`
Example: `myPrefix_CR_CatalogueRequest Link_myPostfix_succeeded_to1WorldSync`

Publication send pattern: `.*CR_CatalogueRequest Publication.*_failed_to1WorldSync`
Example: `myPrefix_CR_CatalogueRequest Publication_myPostfix_failed_to1WorldSync`

Publication send pattern: `.*CR_CatalogueRequest Publication.*_succeeded_to1WorldSync`
Example: `myPrefix_CR_CatalogueRequest Publication_myPostfix_succeeded_to1WorldSync`

File split

Why do we need a file split?

IM, like DSE, allows a file only to contain a certain number of documents (items respectively). The export will process all configured items but a post export step executing an xsl transformation is added at the end. The transformation will split the file every 100 documents and will attach header and footer in order to create a valid file. The files have the same name as configured in the export template but a consecutive number is added at the end.

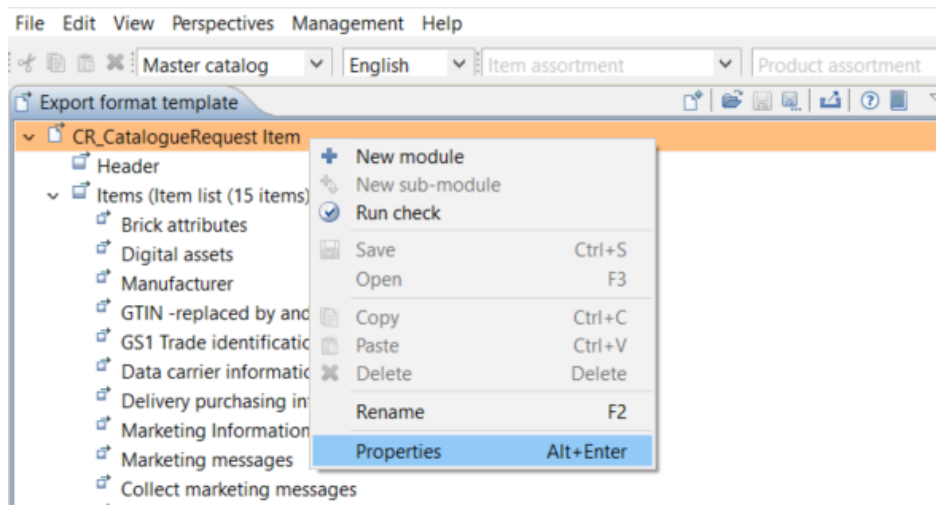
The export templates you find in the resources package already have this post export step attached.

Configuration

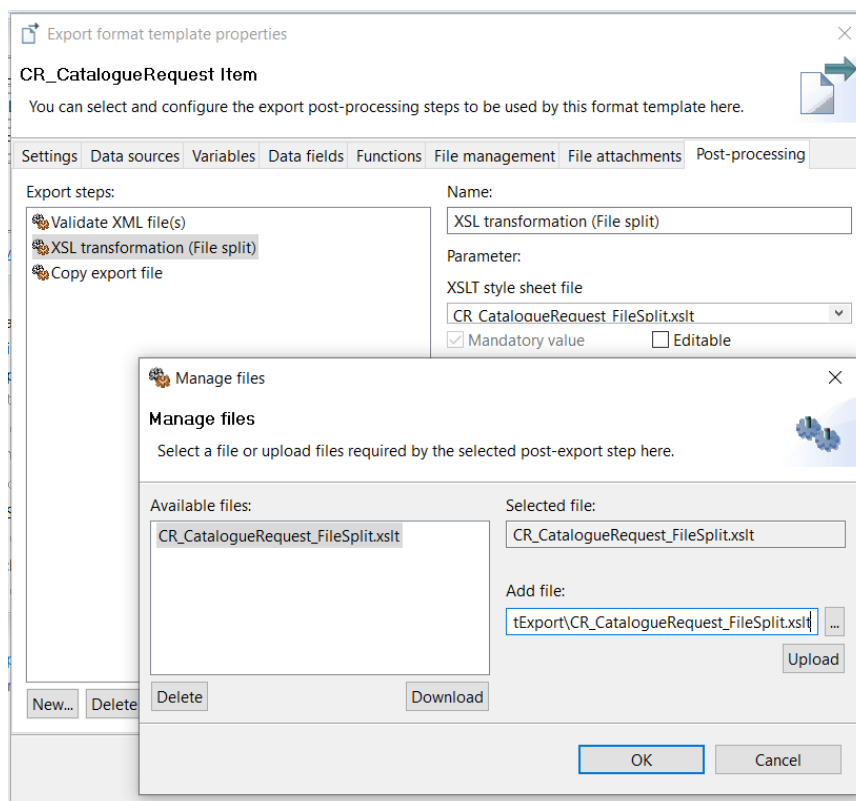
If you create your own export templates or have adjusted one from prior versions, please add the xsl transformation to them.

This is how to configure the export:

1. Open the export template in the *Export format templates* perspective
2. Use a right click on the template to open the *properties* and select the tab *Post-processing*



3. Add a new post export step by selecting *XSL transformation*
You can change the name of the assigned post export step, e.g. XSL transformation (File split)
4. Click on Manage files... to add the stylesheet. The stylesheet can be found in the resources package in the folder DataSource\IM\ExportTemplates\PostExport



Customization

If you need to adjust the number of documents in a file, do the following:

1. Get the stylesheet from the resources package. It can be found in the folder DataSource\IM\ExportTemplates\PostExport and is called "CR_CatalogueRequest_FileSplit.xslt".
2. Adjust the for-each-group tag to use the desired number of documents

Stylesheet	
1	<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" xmlns:n="http://www.1worldsync.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.1worldsync.com http://schemas.preprod.1worldsync.com/schemas/item/2.0/CatalogueRequestProxy.xsd" version="2.0">
2	<xsl:param name="filename"/>
3	<xsl:template match="/n:envelope">
4	<xsl:for-each-group select="catalogueRequest/document" group-adjacent="(position()-1) idiv <Number-of-documents>">

Example	
1	<xsl:for-each-group select="catalogueRequest/document" group-adjacent="(position()-1) idiv 100">

3. Add the changed stylesheet to your export template(s) as described above.

Connection to OpenAS2

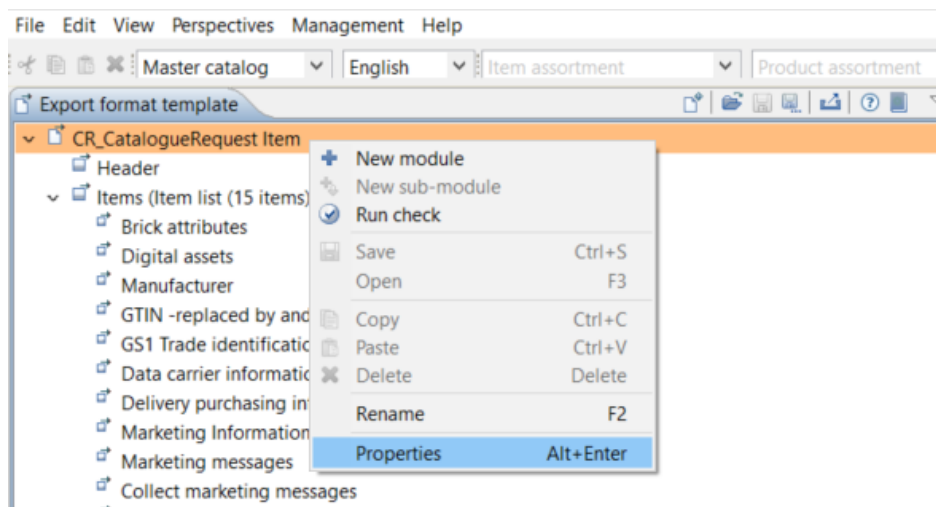
In order to be sent to the GDSN pool, the exported files need to be picked up by OpenAS2. This is done using a hotfolder mechanism. During the installation you defined a hotfolder OpenAS2 listens to. Whenever a file is put in there, it is sent to the defined destination.

The export has a post export step that will copy the export files to the OpenAS2 hotfolder.

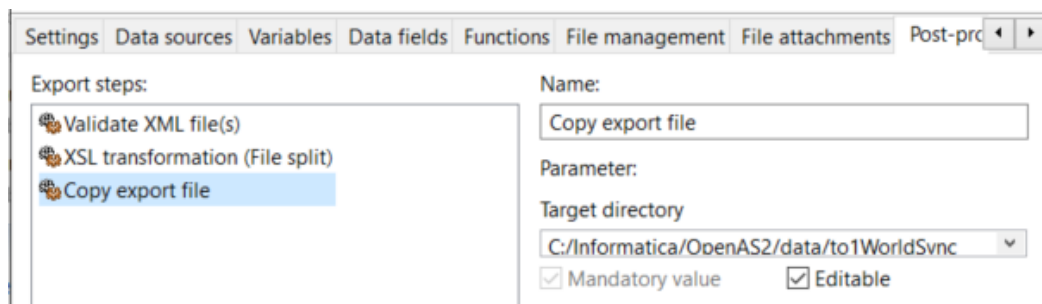
Adjust destination folder

In order to add or adjust the post export step, do the following:

1. See section "Installation / Configuration of OpenAS2" in order to find out which folder you have defined as hotfolder (in the mentioned section, the folder has the name to1WorldSync).
2. In Product360 open the export template in the *Export format templates* perspective
3. Use a right click on the template to open the *Properties* and select the tab *Post-processing*



4. Select or add the post export step *Copy export file*
Important: this export post step has to be the last one
5. Set *Target directory* to the folder you defined as OpenAS2 hotfolder.



2.2.8 Migration Guide

2.2.8.1 Export template migration

File split

In older versions, data transformations in B2B Data Exchange split the files in packages of a certain number of documents the pool can handle. Using OpenAS2 this is now done by a post export step.

Please see [Export/File split](#)(see page 420) on how to configure it.

Copy to hotfolder

The exported files will now be sent to the pool by OpenAS2. This means the export template has to be adjusted to copy the exported files to the OpenAS2 hotfolder instead of the B2B DX ones.

Please see [Export/Connection to OpenAS2](#)(see page 422) on how to change the used hotfolder.

3 Informatica BPM Accelerator

The Informatica BPM Accelerator Package should give some examples helping to start working with BPM in Product 360.

- [Required Workflows](#)(see page 425)
- [Step Workflow](#)(see page 438)
 - [Example 1 \(1 Task\)](#)(see page 458)
 - [Example 2 \(2 Tasks\)](#)(see page 462)
 - [Example 3 \(2 Tasks with DQ checks\)](#)(see page 466)
 - [Example 4 \(5 Tasks with DQ checks, parallel and sequential\)](#)(see page 470)
 - [Example 5 \(extending example 4 with an approval step\)](#)(see page 477)
 - [Example 6 \(2 tasks \(incl. 1 supplier task\) + modifying of fields\)](#)(see page 485)
 - [Example 7 \(2 Tasks + modifying of fields + approval step\)](#)(see page 492)
 - [Example 8 \(2 Tasks modifying of fields approval step merge\)](#)(see page 499)
- [Creating MQ based workflows](#)(see page 504)
- "Required Workflows" contains workflows which are mandatory for working with BPM in the context of Product 360.
- "Step Workflow" contains a best practice example which allows you easily create new workflows for Product 360 based on XML files.
- The chapter "Creating MQ based workflows" describes how to design BPM (ActiveVOS) workflows that leverage the new queue based communication mode between Product 360 and BPM (ActiveVOS).

3.1 Resources

Filename	Belongs to	Description
P360_JMS_Core.bpr	Required Workflows	This BPM workflow is mandatory to support the Active MQ communication between Product 360 server and BPM server.
P360_JMS_Core.zip	Required Workflows	This zip file contains the project resources for the BPM Designer of the above workflow.
P360_BPM_Management.bpr	Required Workflows	This BPM workflow is mandatory to terminate workflows.
P360_BPM_Management.zip	Required Workflows	This zip file contains the project resources for the BPM Designer of the above workflow.
InfNextSteps.bpr	Step Workflow	This bpr file contains mandatory bpm workflows like DQchecks, MergeItems,...
InfNextSteps.zip	Step Workflow	This zip file contains the project resources for the BPM Designer of the above workflow.

Filename	Belongs to	Description
InfaResources.bpr	Step Workflow	This bpr file contains mandatory resources for the Step Workflow like images etc.
InfaResources.zip	Step Workflow	This zip file contains the project resources for the BPM Designer of the above workflow.
StepWorkflow.bpr	Step Workflow	This BPM workflow contains the best practice workflow "StepWorkflow". For more information read the chapter "Step Workflow".
StepWorkflow.zip	Step Workflow	This zip file contains the project resources for the BPM Designer of the above workflow.
StepWorkflowExamples.bpr	Step Workflow	This BPM workflow contains example files for the StepWorkflow.
StepWorkflowExamples.zip	Step Workflow	This zip file contains the project resources for the BPM Designer of the above examples.
P360_JMS_Demo.zip	Creating MQ based workflows	This zip file contains a sample project for the BPM Designer to demonstrate the Queue communication between the servers.
P360_JMS_Demo_DataQuality.zip	Creating MQ based workflows	This zip file contains a sample project for the BPM Designer to demonstrate the Queue communication with DataQuality execution between the servers.

3.2 Required Workflows

It is mandatory to deploy these 2 workflows to ensure an error-free operation of the components Product 360 Server and BPM server.

- The Business Process Archive *P360_BPM_Management.bpr* contains a workflow to terminate workflow process instances provided as parameter. This workflow is needed for the Product 360 "Terminate workflow" functionality and therefore has to be deployed to the Informatica BPM Server. This action is available as workflow task action in the Product 360 web UI as well as in the Desktop client.
- The Business Process Archive *P360_JMS_Core.bpr* contains the workflow which consumes an Active Message Queue and routes the message to the corresponding workflow in the BPM server.

Info

Since there is a dependency between these 2 workflows it is necessary that you the deploy the workflows in the following order:

1. *P360_JMS_Core.bpr*
2. *P360_BPM_Management.bp*

3.2.1 Configuration and Installation of BPM

- Informatica BPM Installation(see page 426)
 - Installation of the Informatica BPM service(see page 426)
 - Webserver and Java(see page 427)
 - Adjusting the webserver to support non SSL port(see page 427)
 - Integrated Security(see page 428)
 - Configuration during installation(see page 428)
 - Re-configuration of already installed server(see page 429)
- Installation of the required default workflows(see page 429)
- Configure Dispatch Services(see page 432)
- Preparing Informatica BPM service for JMS based communication(see page 434)
 - Download additional library(see page 434)
 - ActiveMQ client library(see page 434)
 - Setup messaging service within Informatica BPM(see page 435)
 - Detail configuration of the message manager(see page 435)
 - JMS Messaging Configuration(see page 436)
 - Initial Context Properties(see page 436)
 - Queues & Listeners(see page 437)
 - Verifying the configuration(see page 437)

3.2.2 Informatica BPM Installation

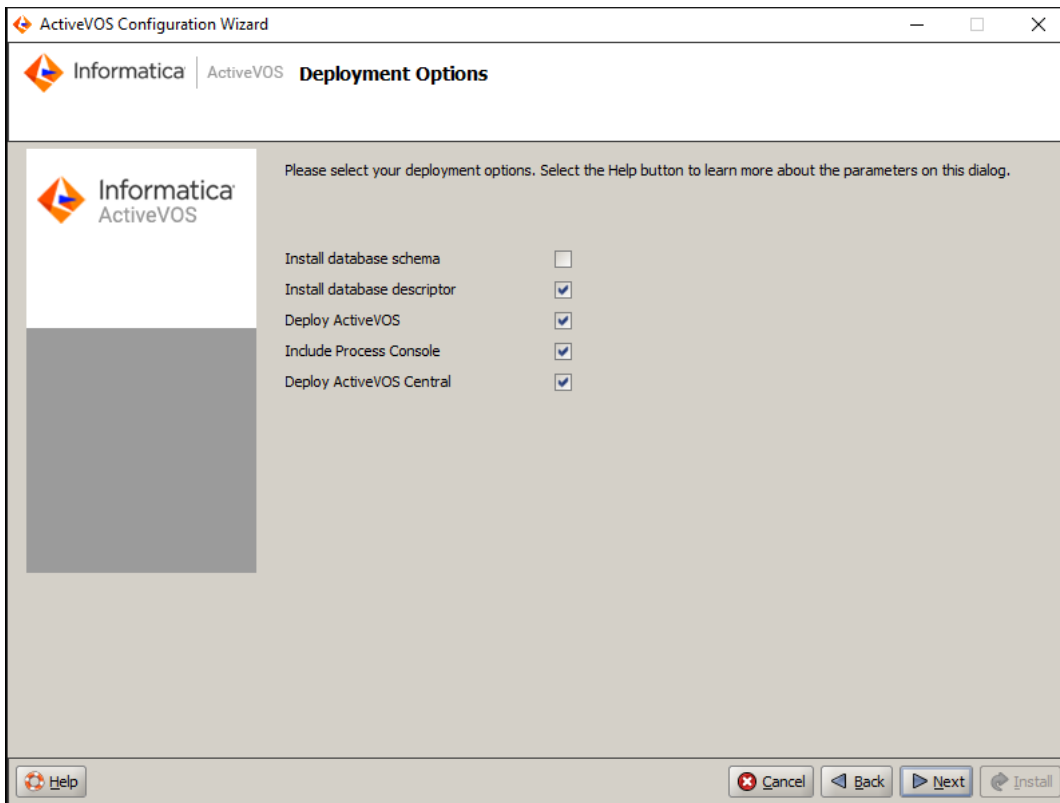
3.2.2.1 Installation of the Informatica BPM service

For the installation of the *Informatica BPM* service it is recommended to follow the official installation guide at <https://docs.informatica.com/process-automation/informatica-activevos/current-version.html>.

In the *Prerequisites* chapter <https://docs.informatica.com/process-automation/informatica-activevos/current-version/4----server-installation--configuration--and-deployment/apache-tomcat/general-information/prerequisites.html> you can find the information about database servers supported for ActiveVOS and additional information about database drivers which are supported. The drivers themselves are not included in the installation package of ActiveVOS and should be get from database vendor.

In the past there were some issues occurred during installation of ActiveVOS if the database schemas have been already created before installation.

So please remember to uncheck "*Install database schema*" if empty DB is already created and to check the "*Install database schema*" option if database schema should be created during installation of ActiveVOS.



3.2.2.2 Webserver and Java

A good start is to use the integration with Apache Tomcat as container where *Informatica BPM service* will be deployed to.

We recommend to use a container version equal or newer to Apache Tomcat 8.5.51.

i Make sure to use Java 8 as runtime for the *Informatica BPM service* by setting the necessary environment variables for example.

Adjusting the webserver to support non SSL port

The Informatica BPM web endpoint does enforce SSL by default. To disable that behavior it is possible to set a JVM property:

```
-Dae.web.filter.https.force=false
```

3.2.2.3 Integrated Security

It's possible to configure the "Informatica BPM" Server to use integrated Security for MS SQL Server connection. In this case the configuration files do not contain sensitive security data like database user and password of the database user.

The configuration can be made during the installation and initial configuration of "Informatica BPM" Server and as a post-configuration for already existing installations.

Configuration during installation

In this case it's better to use the silent installation and configuration mode

1. Create a Windows service user which will be used to execute BPM Server. (e.g. INFA\bpm-service)
2. Create the same user as MSSQL Server user and configure this user to use Windows Authentication (INFA\bpm-service)
3. Create manually a new ActiveVOS database and configure the owner of this database to newly created user (INFA\bpm-service).
4. Install the Webserver (in our case it's Apache Tomcat) and copy the SQL Server driver class and the additional dll (*sqljdbc42.jar* and *sqljdbc_auth.dll*) in the *lib* folder of Tomcat
5. Adapt the *service.bat* to use tomcat lib folder additional to a *java.lib.path* like this: **--JvmOptions "...; ...;-Djava.library.path=%CATALINA_HOME%\lib"**
6. Extract the installation and configuration tool for "Informatica BPM".
Go to the `<installation_tool>\server-enterprise\tomcat_config\bin` folder and adapt the *install.properties* for server configuration. See example for properties relevant to integrated security. The content of username and password is mandatory but not relevant for the connection. So you can use any signs.

Properties

```
jdbc.database.driver.class=com.microsoft.sqlserver.jdbc.SQLServerDriver
jdbc.database.driver.jar=<tomcat_path>\lib\sqljdbc42.jar
jdbc.database.url=jdbc:sqlserver://
<sqlserver_host>;databaseName\=Active_VOS;integratedSecurity=true;
jdbc.database.name=Active_VOS
jdbc.database.password=xxxx
jdbc.database.username=xxxx
```

7. Go to the `<installation_tool>\server-enterprise\tomcat_config\bin` folder and adapt the *config_deploy.bat* to use tomcat lib folder additional to a *java.lib.path* like this:

Configuration

```
"<jdk_path>\bin\java" -Xms128m -Xmx512m -Djava.library.path="<tomcat_path>\lib"
-jar config.jar %1
```

8. Open the command window and execute the *config_deploy.bat* in **silent** mode.
9. Install the Tomcat service using *service.bat install*

10. Configure the "Log on" for this service to use the BPM service account (INFA\bpm-service)
11. Start the service and call the ActiveVOS Console.

Re-configuration of already installed server

Following steps can be made to re-configure the existing BPM Server installation to use the integrated security for database connection:

1. Stop and uninstall the Tomcat service for BPM Server. Use *service.bat uninstall [tomcat service name]*
2. Create a Windows service user which will be used to execute BPM Server. (e.g. INFA\bpm-service)
3. Create the same user as MSSQL Server user and configure this user to use Windows Authentication (INFA\bpm-service)
4. Configure the owner of the existing ActiveVOS database to newly created user (INFA\bpm-service).
5. Go to the webserver (Tomcat) installation ({install_dir}/apache-tomcat/conf/Catalina) and to the BPM server ({install_dir}/server-enterprise/tomcat_config/conf) and adapt activevos.xml and active-bpel.xml in both places to use integrated security (s. example). The content of username and password is mandatory but not relevant for the connection. So you can use any signs like in example.

Configuration

```
<Context displayName="ActiveBPEL Enterprise Tomcat Database context" path="/
active-bpel">
  <Resource name="jdbc/ActiveVOS" auth="Container" type="javax.sql.DataSource"
    maxActive="100"
    maxIdle="10"
    maxWait="1000"
    username="xxxx"
    password="xxxx"
    driverClassName="com.microsoft.sqlserver.jdbc.SQLServerDriver"
    url="jdbc:sqlserver://
server.informatica.com;databaseName=Active_VOS;integratedSecurity=true;"/>
</Context>
```

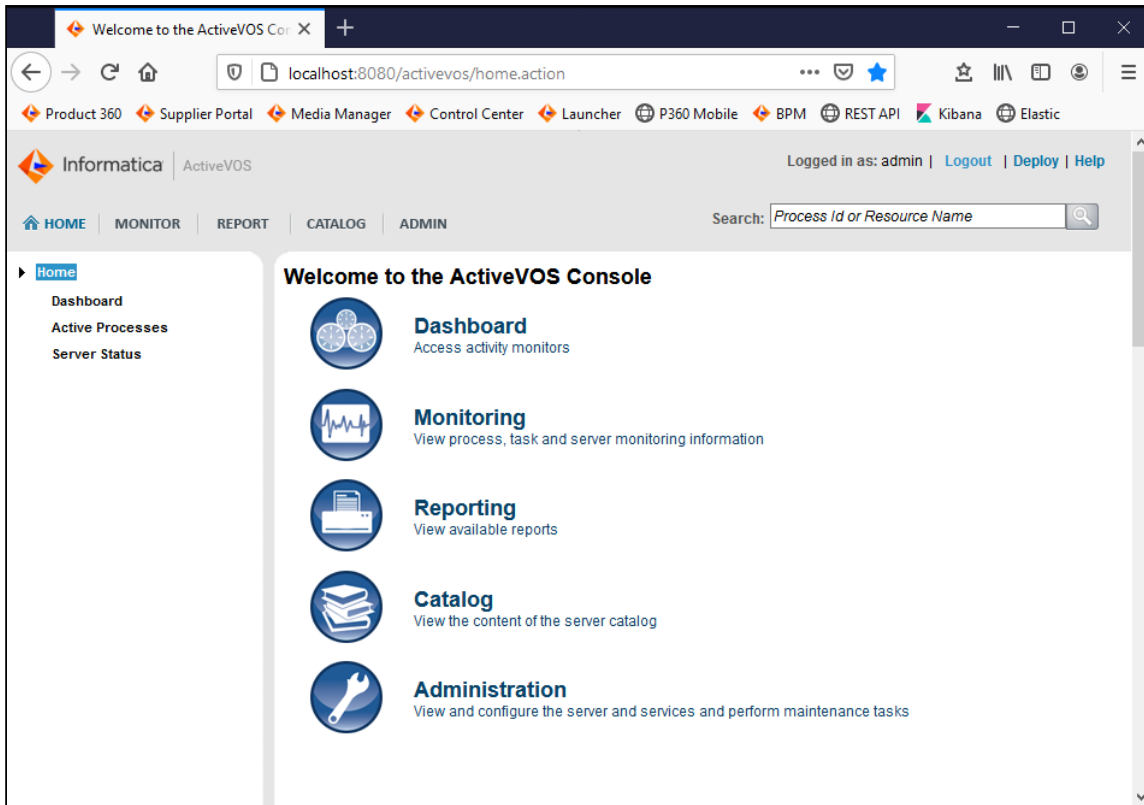
6. Copy the driver class and the additional dll (*sqljdbc42.jar* and *sqljdbc_auth.dll*) in the lib folder of Tomcat
7. Adapt the *service.bat* to use tomcat lib folder additional to a java.lib.path like this: **--JvmOptions "...; ...;-Djava.library.path=%CATALINA_HOME%\lib"**
8. Install the Tomcat service using *service.bat install*
9. Configure the "Log on" for this service to use the BPM service account (INFA\bpm-service)
10. Start the service and call the ActiveVOS Console.

3.2.3 Installation of the required default workflows

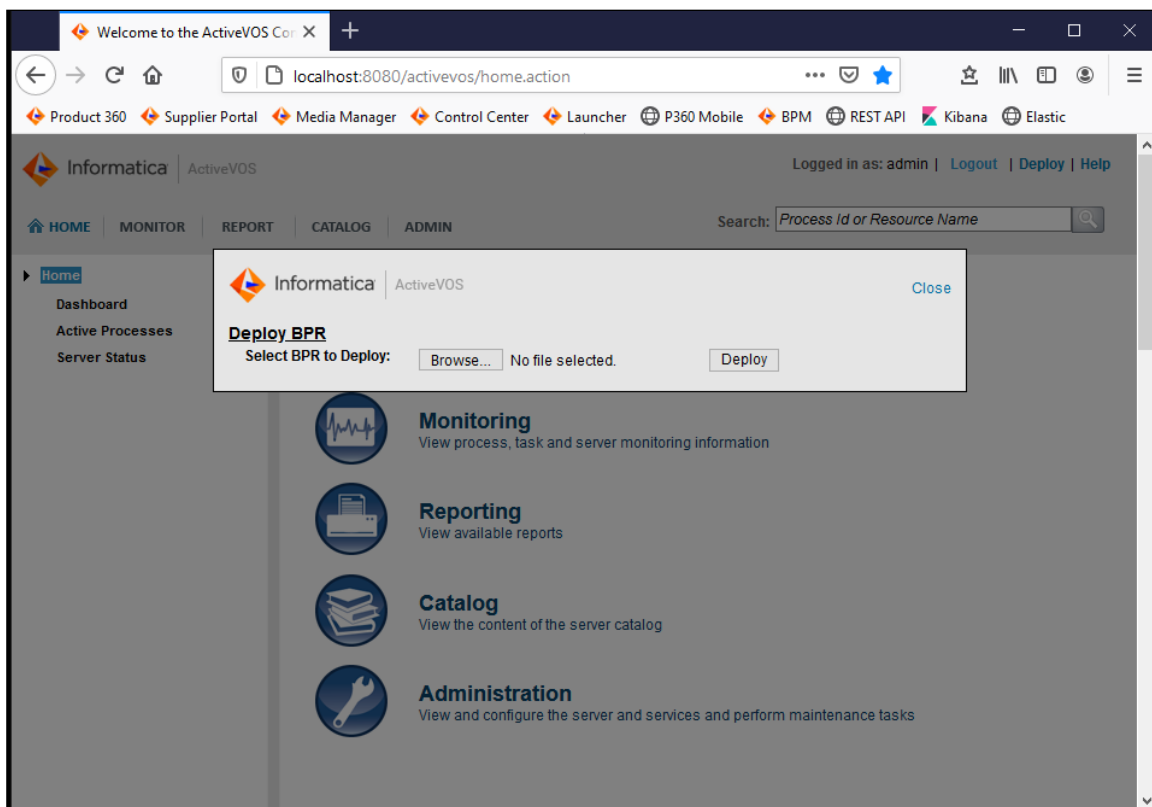
After installing the *Informatica BPM* service the required default workflows have to be deployed to the BPM instance.

The workflows are provided as deployable Business Process Archives: **P360_JMS_Core.bpr** and **P360_BPM_Management.bpr** in the Accelerator package <P360 version>_InformaticaBPM of the P360 release.

To deploy the workflows make sure the *Informatica BPM service* is up and running: Open your web browser and open the management console located at <http://your-bpm-server:8080/activevos>. You should see the start page of the management console.



To deploy the workflows open the deploy dialog by clicking on the "Deploy" button in the upper right corner and upload the Business Process Archives.



After the upload and deployment has been processed successfully you can check the availability of the P360 default workflows by navigating to *Catalog/Contributions* where you should now see the deployed projects containing the P360 default workflows in the list.

Welcome to the ActiveVOS Console

localhost:8080/activevos/contribution_listing.action

Product 360 | Supplier Portal | Media Manager | Control Center | Launcher | P360 Mobile | BPM | REST API | Kibana | Elastic

Informatica | ActiveVOS | Logged in as: admin | Logout | Deploy | Help

HOME | MONITOR | REPORT | CATALOG | ADMIN

Search: Process Id or Resource Name

Catalog

- Contributions
- Process Definitions
- Indexed Properties
- Partner Definitions
- Service Definitions
- Task Properties
- Resources
 - All
 - Central Configs
 - Function Contexts
 - HTML Documents
 - Images
 - Java Jars
 - Report Definitions
 - Schema Documents
 - WSDL Documents
 - XQuery Modules
 - XSL Documents
 - Other

Contributions

Contribution	Version	State	Date	Group	Deployer
project/InfResources	1.0	ONLINE	2020/11/25 09:07 PM		admin
project/P360_BPM_Management	1.0	ONLINE	2020/11/25 09:07 PM		admin
project/P360_JMS_Core	1.0	ONLINE	2020/11/25 09:07 PM		admin
project/StepWorkflow	1.0	ONLINE	2020/11/25 09:07 PM		admin
project/StepWorkflowExamples	1.0	ONLINE	2020/11/25 09:07 PM		admin

20 records per page. Results 1 - 5 of 5

Selection Filter

State: Online

Deployed between: and (yyyy/mm/dd)

Name:

Group: System ☒ Hide

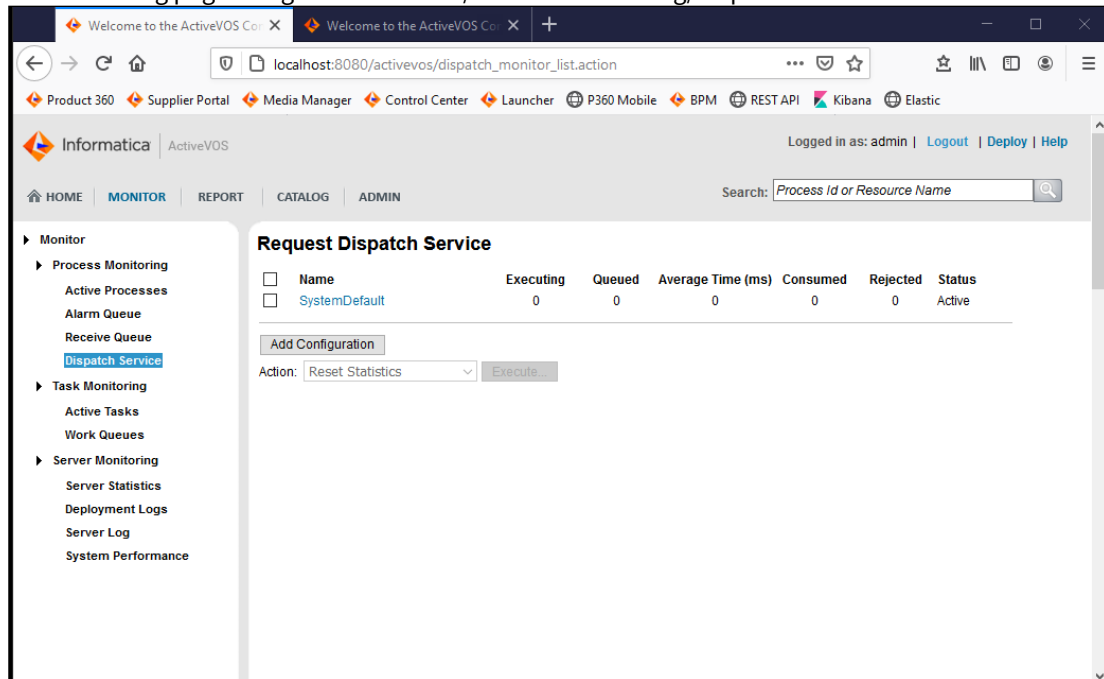
Submit Clear

3.2.4 Configure Dispatch Services

The dispatch services have to be configured in the console of BPM:

1. Open a web browser and navigate to <http://your-bpm-server:8080/activevos> and login using your user credentials

- On the landing page navigate to "Monitor/Process Monitoring/Dispatch Service"



- Click on SystemDefault and update the values

Dispatch Configuration

Name: SystemDefault

Max Concurrent: 150

Max In-Memory: 20000

Max Queued: 20000

Timeout (seconds): 300

Persistent: ☒

At runtime, the dispatch configuration used for a particular request is chosen based on matching the configuration name in the following order of precedence: Service Name, Process Group, Tenant, System Default

Dispatch Configuration	
Name	SystemDefault
Max Concurrent	150
Max in Memory	20000
Max Queued	20000
Timeout (seconds)	300

Dispatch Configuration	
Persistent	true

4. Click on Update Configuration
5. Click on Add Configuration and set the values

Dispatch Configuration

Name: P360RouterService

Max Concurrent: 100

Max In-Memory: 5000

Max Queued: 5000

Timeout (seconds): 300

Persistent: ☒

Update Configuration Close

At runtime, the dispatch configuration used for a particular request is chosen based on matching the configuration name in the following order of precedence: Service Name, Process Group, Tenant, System Default

Dispatch Configuration	
Name	P360RouterService
Max Concurrent	100
Max in Memory	5000
Max Queued	5000
Timeout (seconds)	300
Persistent	true

6. Click on Update Configuration

3.2.5 Preparing Informatica BPM service for JMS based communication

The communication between Informatica Product 360 and Informatica BPM service can be switched to an asynchronous message based communication. To enable this capability of Informatica BPM service some post installation steps have to be performed.

3.2.5.1 Download additional library

ActiveMQ client library

To make Informatica BPM service ready for working with messaging an additional library has to be installed.

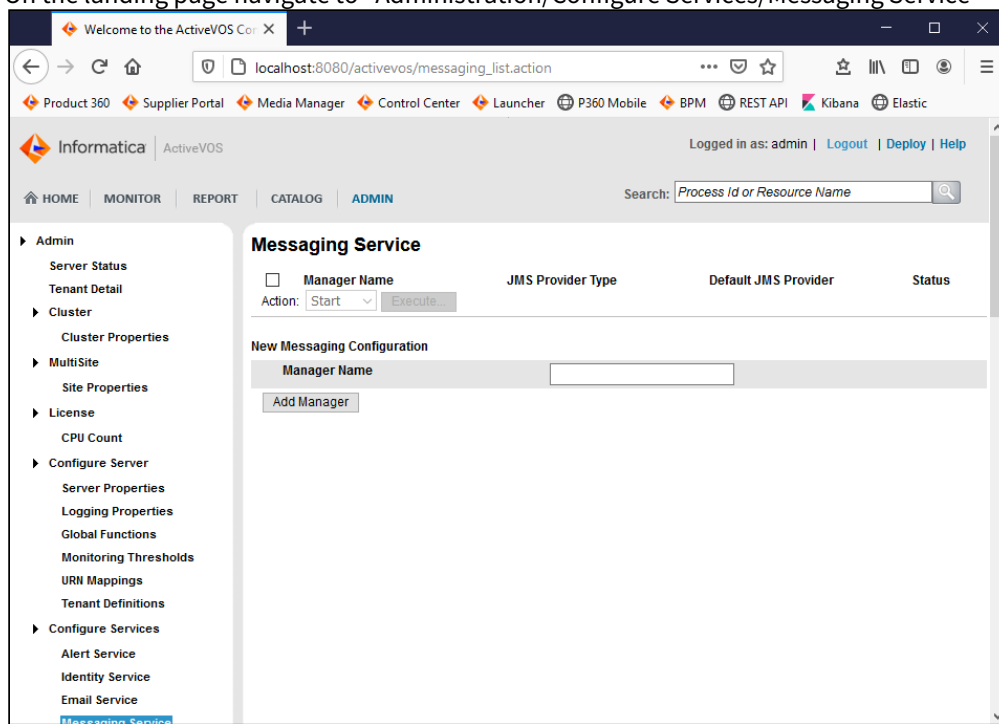
- Download <https://repo1.maven.org/maven2/org/apache/activemq/activemq-all/5.15.9/activemq-all-5.15.9.jar>
- Put the downloaded file `activemq-all-5.15.9.jar` into the `lib` folder of the Tomcat container
- Ensure the JDK version is ≥ 1.8 as the active mq lib requires that

Make sure to restart the Informatica BPM service so that the changes take effect.

3.2.5.2 Setup messaging service within Informatica BPM

To be able to use the asynchronous message based communication between Informatica Product 360 and Informatica BPM service some basic configuration within Informatica BPM has to be set up. This configuration will be done using the Informatica BPM management console.

1. Open a web browser and navigate to `http://your-bpm-server:8080/activevos` and login using your user credentials
2. On the landing page navigate to "Administration/Configure Services/Messaging Service"



3. Create a new "Manager" entering "ActiveMQ" as name and clicking on the "Add Manager" button
4. A new entry with name "ActiveMQ" will appear in the list of manager, to configure the manager in detail click on the manager entry

Detail configuration of the message manager

To finish the configuration of the manager you have to add the following values as minimum configuration

JMS Messaging Configuration

Property name	Property value
Default JMS Provider	enabled
JMS Provider Type	Other JMS
Connection Factory Name	ConnectionFactory
Connection User	The username to connect to the ActiveMQ message broker
Connection Password	The password to connect to the ActiveMQ message broker
Send Empty Credentials	disabled
Maximum Total Connections	-1
Maximum Free Connections	15
Delivery Mode	Persistent
Time To Live (ms)	0
Priority (int)	0
Reconnect Interval (ms)	

Initial Context Properties

Property name	Property value
java.naming.provider.url	tcp://your-activemq-server:61616? jms.redeliveryPolicy.maximumRedeliveries=-1 ¹
java.naming.factory.initial	org.apache.activemq.jndi.ActiveMQInitialContextFactory
queue.JNDI_P360_BPM	P360_BPM
queue.JNDI_P360_SERVICE_API	P360_SERVICE_API
queue.JNDI_P360_BATCH_API	P360_BATCHAPI
queue.JNDI_P360_BPM_RESPONSE	P360_BPM_RESPONSE

¹ if using ssl connection make sure to import the corresponding certificates into your JREs trust store.

Queues & Listeners

Add a new entry in section Queues & Listeners, not mentioned properties need to stay with their default value

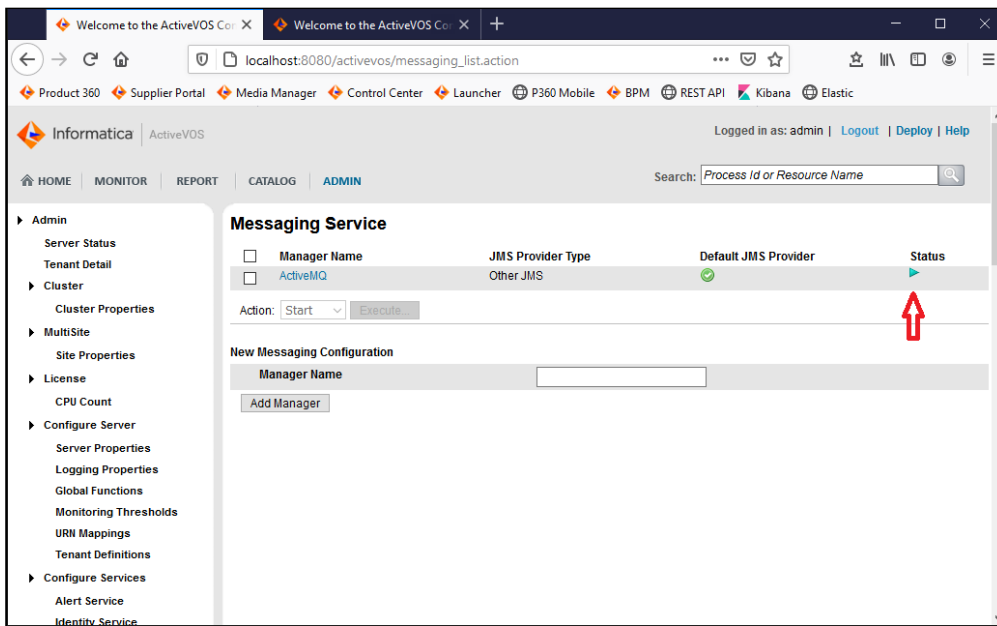
Property name	Property value
Name	P360_BPM_LISTENER
JNDI Location	JNDI_P360_BPM
Listener Class	com.activeee.rt.mom.jms.transport.AeJmsBpelListener
Listener Count	15
Default Service	P360RouterService

Add a new entry in section Queues & Listeners, not mentioned properties need to stay with their default value

Property name	Property value
Name	P360_BPM_RESPONSE_LISTENER
JNDI Location	JNDI_P360_BPM_RESPONSE
Listener Class	com.activeee.rt.mom.jms.transport.AeJmsBpelListener
Listener Count	15
Default Service	P360RouterService

Verifying the configuration

After the manager configuration has been validated and saved you can go back to the Messaging Service overview page. Your manager should now appear in the list with status running.



3.3 Step Workflow

3.3.1 Preface

The StepWorkflow project is a workflow with the following objectives:

- You don't need to be a BPM (AVOS) expert to design workflows for MDM-Product 360 business use cases
- Development of new workflows is more structured (in comparison to previous developments)
- Easier to troubleshoot
- Less issues because of tested re-usable code
- Reduce the amount of running processes on the BPM server
- Enable batching

Table of Contents

- [Preface\(see page 438\)](#)
- [Prerequisites\(see page 439\)](#)
- [Installation\(see page 439\)](#)
- [Configuration\(see page 439\)](#)
- [StepWorkflow XML files\(see page 440\)](#)
 - [The StepWorkflowMap.xml file\(see page 441\)](#)
 - [The StepWorkflow.xml file\(see page 443\)](#)
- [Examples\(see page 446\)](#)
 - [Steps to run the examples\(see page 446\)](#)
 - [Example 1 \(1 Task\)\(see page 448\)](#)
 - [Example 2 \(2 Tasks\)\(see page 448\)](#)
 - [Example 3 \(2 Tasks with DQ checks\)\(see page 448\)](#)
 - [Example 4 \(5 Tasks with DQ checks, parallel and sequential\)\(see page 448\)](#)
 - [Example 5 \(extending example 4 with an approval step\)\(see page 448\)](#)
 - [Example 6 \(2 tasks \(incl. 1 supplier task\) + modifying of fields\)\(see page 448\)](#)

- [Example 7 \(2 Tasks + modifying of fields + approval step\)\(see page 448\)](#)
- [Example 8 \(2 Tasks modifying of fields approval step merge\)\(see page 448\)](#)
- [Example 9 \(Classic task \)\(see page 448\)](#)
- [Defaults\(see page 453\)](#)
 - [DefaultArticle\(see page 453\)](#)
 - [DefaultVariant\(see page 455\)](#)
 - [DefaultProduct2G\(see page 457\)](#)

3.3.2 Prerequisites

The StepWorkflow is designed for the queue communication. Therefor Product 360 must also be configured for Queue communications. Please reach out to the Installation and Operation guide for the following areas:

- Configuration of queues in the Product 360 server
- Installation of ActiveVOS with ActiveMQ support and configuration of queues in ActiveVOS (*Note: ActiveVOS version 9.2.4.6 is a prerequisite for queue based interactions between Product 360 and ActiveVOS*)
- Message headers and message formats for the various interactions between Product 360 and ActiveVOS

3.3.3 Installation

Before you can design your workflows based on the StepWorkflow you have to deploy these 4 resources:

1. InfaResources.bpr
2. StepWorkflow.bpr
3. InfaNextSteps.bpr
4. StepWorkflowExamples.bpr (optional)

with the BPM (AVOS) console.

Also it is possible that you deploy the workflows with your BPM designer. The resources of the projects are in the zip files:

1. InfaResources.zip
2. StepWorkflow.zip
3. InfaNextSteps.zip
4. StepWorkflowExamples.zip

3.3.4 Configuration

The base configuration of the StepWorkflow is defined in the BPM URN mappings. Therefore you have to define the following mappings in the BPM console.

URN	URL	Description
urn:p360.api.username	RestServiceUser	Name of the Product 360 user in which context the requests are fired
urn:p360.api.password	*****	Password of the user above

URN	URL	Description
urn:p360.api.manager	ActiveMQ	Name of the configured messaging service of the BPM server
urn:p360.api.queue	JNDI_P360_SERVICE_API	JNDI name of the queue for the service calls
urn:p360.rest.url	http://p360server:1512	Url of the Product 360 rest services
urn:dq.timeout	PT4H	Timeout for DQ calls
urn:p360.api.data.quality.queue	JNDI_P360_DATA_QUALITY	JNDI name of the queue for the Data Quality calls
urn:p360.api.data.quality.response.queue	bpm_response	Identifier if the response queue for DQ calls (default: bpm)
urn:list.get.timeout	PT4H	Timeout for list calls
urn:merge.delay	PT30S	Delay before starting the merge process
urn:error.task.group	Superuser	Identifier of the usergroup for which a classic task will be created if an error happens within the StepWorkflow. (i.e. Step Id is not defined)
urn:workflow.resource.project	StepWorkflowExamples	Name of the project (AVOS catalog) where the Mapfile is stored

3.3.5 StepWorkflow XML files

The StepWorkflow itself contains sever xml files in the catalog resources.

Name	Description
StepWorkflowMap.xml	This xml file defines the mapping between a Product 360 trigger configuration, Product 360 workflow identifier and a stepworkflow.xml file.
StepWorkflow.xml	Such a file defines the steps for a specific business workflow. The name of the file should be adapted to your use case. A valid example can be ItemCreation.xml.
Manufacturer Teams.xml	This xml file defines the mapping between Product 360 usergroups and Product 360 suppliers (manufacturers). This is a showcase how to create tasks to different groups based on the manufacturer.

3.3.5.1 The StepWorkflowMap.xml file

Tag	Description
files	Within the files tag are the different file mappings that need to be defined
file	Each business workflow definition needs to be defined in this tag.
path	Path and name of the workflow. (Case-sensitive!)
triggerConfiguration	Name of the Product 360 trigger configuration.
workflowIdentifier	Identifier of the Product 360 workflow.

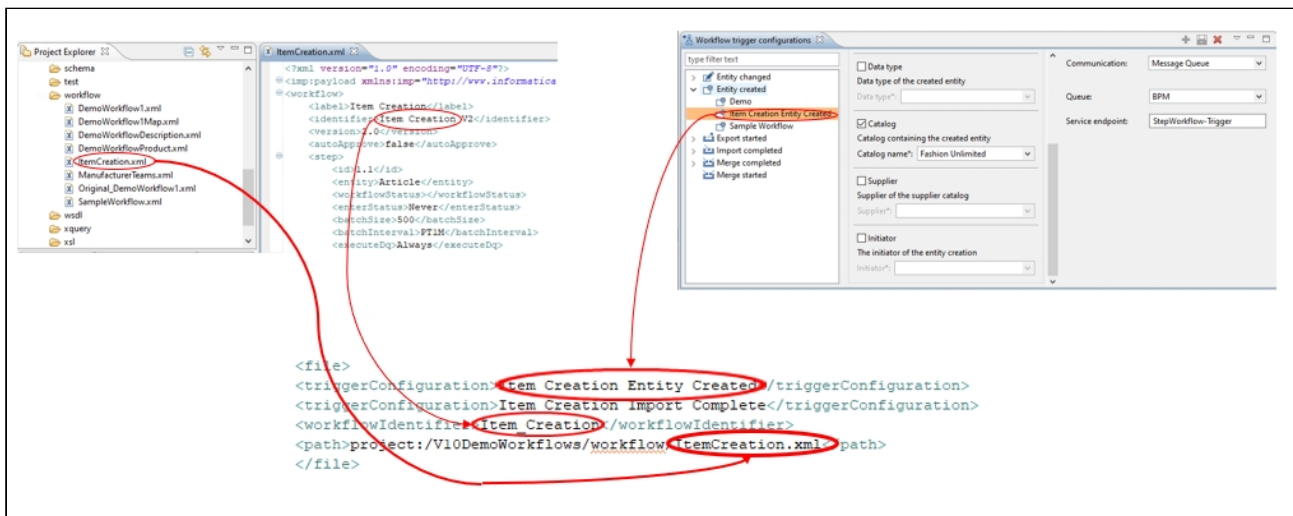
StepWorkflowMap

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <files>
    <file>
      <workflowIdentifier>Default_Workflow_Article</workflowIdentifier>
      <path>project:/StepWorkflow/workflow/DefaultArticle.xml</path>
    </file>
    <file>
      <workflowIdentifier>Default_Workflow_Product2G</workflowIdentifier>
      <path>project:/StepWorkflow/workflow/DefaultProduct2G.xml</path>
    </file>
    <file>
      <workflowIdentifier>Default_Workflow_Variant</workflowIdentifier>
      <path>project:/StepWorkflow/workflow/DefaultVariant.xml</path>
    </file>
    <file>
      <triggerConfiguration>Example01</triggerConfiguration>
      <workflowIdentifier>Workflow_01</workflowIdentifier>
      <path>project:/StepWorkflow/workflow/Workflow_01.xml</path>
    </file>
    <file>
      <triggerConfiguration>Example02</triggerConfiguration>
      <workflowIdentifier>Workflow_02</workflowIdentifier>
      <path>project:/StepWorkflow/workflow/Workflow_02.xml</path>
    </file>
    <file>
      <triggerConfiguration>Example03</triggerConfiguration>
      <workflowIdentifier>Workflow_03</workflowIdentifier>
      <path>project:/StepWorkflow/workflow/Workflow_03.xml</path>
    </file>
  </files>
</imp:payload>
```

```

</file>
<file>
  <triggerConfiguration>Example04</triggerConfiguration>
  <workflowIdentifier>Workflow_04</workflowIdentifier>
  <path>project:/StepWorkflow/workflow/Workflow_04.xml</path>
</file>
<file>
  <triggerConfiguration>Example05</triggerConfiguration>
  <workflowIdentifier>Workflow_05</workflowIdentifier>
  <path>project:/StepWorkflow/workflow/Workflow_05.xml</path>
</file>
<file>
  <triggerConfiguration>Example06</triggerConfiguration>
  <workflowIdentifier>Workflow_06</workflowIdentifier>
  <path>project:/StepWorkflow/workflow/Workflow_06.xml</path>
</file>
<file>
  <triggerConfiguration>Example07</triggerConfiguration>
  <workflowIdentifier>Workflow_07</workflowIdentifier>
  <path>project:/StepWorkflow/workflow/Workflow_07.xml</path>
</file>
<file>
  <triggerConfiguration>Example08</triggerConfiguration>
  <workflowIdentifier>Workflow_08</workflowIdentifier>
  <path>project:/StepWorkflow/workflow/Workflow_08.xml</path>
</file>
<file>
  <triggerConfiguration>Example09</triggerConfiguration>
  <workflowIdentifier>Workflow_09</workflowIdentifier>
  <path>project:/StepWorkflow/workflow/Workflow_09.xml</path>
</file>
<file>
  <triggerConfiguration>Example10</triggerConfiguration>
  <workflowIdentifier>Workflow_10</workflowIdentifier>
  <path>project:/StepWorkflow/workflow/Workflow_10.xml</path>
</file>
</files>
</imp:payload>

```



3.3.5.2 The StepWorkflow.xml file

The separate StepWorkflow.xml files define the different business workflows. In this chapter we will describe, and provide examples for different workflows. If you want to create new xml files within your workflow you have to open the StepWorkflow project in BPM designer and add a new xml file in the workflow folder with the tag `imp:payload` to ensure that this file will be deployed with your workflow. After that, you have to register this new xml file in the StepWorkflowMap.xml file.

Description of the StepWorkflow.xml file

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <!-- This is a description and sample values of the fields in the Workflow Detail
  files and should not be used directly
  You should not use xml files with a lot of comments for live processing. Please
  remove all comments in your
  Workflow Detail files -->
  <workflow>
    <label>My Stepworkflow</label>
    <!-- Name of the workflow that is visible in the UI -->
    <identifier>My_Stepworkflow_V1</identifier>
    <!--Name of the workflow not visible in the UI must be unique thru all workflows
    -->
    <version>1.0</version>
    <!--version number of the workflow needs to be incremented by one if a Status,
    Usergroup, description, decision or
    UiTemplate has been changed -->
    <step>
```

```
<!-- These are the steps to the workflow. They do not have to include entering
into a task -->
```

```
<id>1.1</id>
```

```
<!-- This is the identifier of each step. They must be unique within a
workflow.
```

```
While generally numeric in nature, can be any string value. -->
```

```
<entity>Article</entity>
```

```
<!-- This is the entity container that the items will be placed in
within the workflow in Product 360. Valid values are:
```

```
Article
```

```
Variant
```

```
Product2G -->
```

```
<workflowStatus>StartCE01</workflowStatus>
```

```
<!-- This will be the status name within the Product 360 workflow. It will be
the second part of the name of the task name in Product 360:
```

```
Product 360 task name = [Workflow] - [Status] - [Catalog] -->
```

```
<description>Desc 1.1</description>
```

```
<!-- Description of the status. It will be displayed in the task in the
Product 360 UI -->
```

```
<enterStatus>Never</enterStatus>
```

```
<!-- The circumstances when an item should enter a status (task):
```

```
OnDqResults=Item(s) will enter the task/status on DQ failures or move to the
next step on successfuls.
```

```
Never=This is for steps that do not include a task
```

```
Always=Item(s) will always enter a task/status the first time the step is
run. Subsequent times are based on the dq results. -->
```

```
<batchSize>500</batchSize>
```

```
<!-- This is the batch size to use with trigger batching. If the triggers
coming from Product 360 have a large number of entities,
this can break them down. Batch sizes can make Product 360 requests more
efficient -->
```

```
<userType>userGroup</userType> <!-- Type of the default assignee valid
values:
```

```
user
```

```
userGroup
```

```
supplier ==> mandatory field: <getField>Article.MainSupplier->Party.Name</
getField> has to be set!-->
```

```
<userName>Standardusers</userName>
```

```
<!--Default assignee of the task. Should be either a Product 360 user or
userGroup -->
```

```
<uiTemplate>Item approve UI</uiTemplate>
```

```
<!-- default UI of the task -->
```

```
<singleChoice>true</singleChoice> <!-- This is for Rejects only and
determines if you can have more than one choice. Should be true or nothing as it
defaults to 'false' -->
```

```
<rejectDecision>
```

```
<id>p360.bpm.reject.TRIGGER:4.1</id> <!-- Always starts with
'p360.bpm.reject.' Can continue with:
```

```
STEP : Which step is rejected
```

```
ex. p360.bpm.reject.STEP:5.1
```

```
TRIGGER : Which parallel task is
```

```
rejected ex. p360.bpm.reject.TRIGGER:4.1
```


SERVICE : Which process is generated with the reject ex. p360.bpm.reject.TaskByManufacturer-Workflow -->

```

    <label>Maintain texts</label>    <!-- The label of the task being rejected.
-->
    </rejectDecision>
    <executeDq>Always</executeDq>
    <!-- The circumstances when a DQ should be run:
        Always=DQ will be executed whenever this step is run regardless of whether it
        was run as a next step or from a trigger.
        OnTrigger=DQ will run only when the process was started from a trigger in
        Product 360 and not from a previous step.
        Never=DQ is not used for this step. -->
    <dqService>ExecuteDqBatch-Process</dqService>
    <!-- The service name of the process that will be handle the processing of the
    dq or any custom built business rules. -->
    <dqRule>Item image check</dqRule>
    <!-- List any Product 360 DQ rules you want to have run with a step. You can
    have multiple entries. -->
    <dqRuleGroup>Item descriptions</dqRuleGroup>
    <!-- List any Product 360 DQ ruleGroups you want to have run with a step. You
    can have multiple entries. -->
    <dqChannel>01 Classify</dqChannel>
    <!-- List any Product 360 DQ channels you want to have run with a step. You
    can have multiple entries. -->
    <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
    <!-- This field allows the step to update a field in Product 360. The field
    must be fully qualified. -->
    <updateFieldValue>600</updateFieldValue>
    <!-- This is the value for the update. -->
    <getField>Article.CurrentStatus</getField>
    <!-- List of any fields directly on the entity that will be required. These
    entries must be fully qualified. -->
    <nextStep>STEP:3.1</nextStep>
    <!-- This is the step that the item(s) will move to if dq is successful or
    there is no dq.
        It must start with STEP: to move to a step. Otherwise it will be interpreted
        as a BPEL service that you can use for custom code.
        In both cases a the item(s) are sent in list using the ItemMap schema. -->
    <dqFailStep>TaskByManufacturer-Workflow</dqFailStep>
    <!-- This is the step that the item(s) move to if dq fails It must start with
    STEP: to move to a step.
        Otherwise it will be interpreted as a BPEL service that you can use for
        custom code.
        In both cases a the item(s) are sent in list using the ItemMap schema. -->
    <nextStep>StepWorkflow_ParallelTasks-Workflow</nextStep>
    <!-- StepWorkflow_ParallelTasks-Workflow should be used for steps that run in
    parallel -->
    <dqFailStep>StepWorkflow_ParallelTasks-Workflow</dqFailStep>
    <!-- StepWorkflow_ParallelTasks-Workflow should be used for steps that run in
    parallel -->
    <parallelStep>4.1</parallelStep>

```

```

<!-- This is where you list parallel steps if a step should go to multiple
steps at the same time -->
<parallelNextStep>5.1</parallelNextStep>
<!-- If all the parallel steps finish, then the item will move to this step -->
<dqFailTrigger>StepWorkflow_ParallelTasks-Finish-3</dqFailTrigger>
<!-- Used to route to the Parallel steps after logic in StepWorkflow process is
finished. -->
<nextTrigger>StepWorkflow_ParallelTasks-Finish-3</nextTrigger>
<!-- Used to route to the Parallel steps after logic in StepWorkflow process is
finished. -->
</step>
</workflow>
</imp:payload>

```

3.3.6 Examples

3.3.6.1 Steps to run the examples

- Check the table "**Prerequisites for Product 360**" and prepare your system accordingly.
- Open the BPM console in your web browser and navigate to Catalog/Contributions/project:/StepWorkflowExamples and click on it.

The screenshot shows the Informatica ActiveVOS Catalog interface. The top navigation bar includes 'HOME', 'MONITOR', 'REPORT', 'CATALOG', and 'ADMIN'. The 'CATALOG' tab is selected. On the left, a sidebar lists various catalog items, with 'Contributions' expanded. The main area displays a table of Contributions with columns: Contribution, Version, State, Date, Group, and Deployer. The table lists six contributions, all in an 'ONLINE' state, deployed by 'admin'. Below the table, there are filters for 'State' (set to 'Online') and 'Deployed between' (with date pickers). A 'Selection Filter' section includes fields for 'Name' and 'Group', and a 'Hide System' checkbox. 'Submit' and 'Clear' buttons are at the bottom right of the filter section.

Contribution	Version	State	Date	Group	Deployer
project/ContentEnrichmentWorkflow	2.0	ONLINE	2020/08/03 11:54 PM		admin
project/InfraResources	1.0	ONLINE	2020/10/25 10:52 PM		admin
project/P360_BPM_Management	1.0	ONLINE	2020/10/25 10:52 PM		admin
project/P360_JMS_Core	1.0	ONLINE	2020/10/25 10:52 PM		admin
project/StepWorkflow	2.0	ONLINE	2020/10/27 04:30 AM		admin
project/StepWorkflowExamples	7.0	ONLINE	2020/10/27 07:07 AM		admin

- Select one of the png files and click on it to see the flow of the workflow.

The screenshot shows the Informatica ActiveVOS Catalog interface. The left sidebar contains a navigation menu with categories like Contributions, Process Definitions, Indexed Properties, Partner Definitions, Service Definitions, Task Properties, and Resources. The main area displays the 'Contribution Detail' for a contribution with ID 358. The details include Contribution: project:/StepWorkflowExamples, Version: 7.0, State: ONLINE, and Group: . Below this, there are two tables: 'Deployed Processes' and 'Contributed Resources'.

Contribution Detail

Id: 358
 Contribution: project:/StepWorkflowExamples
 Version: 7.0
 State: ONLINE
 Group:
 Description:
 [Set to Offline] [Delete...]

Deployed Processes				
Name	Target Namespace	Version	State	Group
Contributed Resources				
Name	Version	Online	Target Namespace	Group
Example_01 (1 Task).png	1.0.3	Yes		
Example_01.xml	1.0.8	Yes		
Example_02 (2 Tasks).png	1.0.3	Yes		
Example_02.xml	1.0.8	Yes		
Example_03 (2 Tasks with DQ checks).png	1.0.3	Yes		
Example_03.xml	1.0.8	Yes		
Example_04 (5 Tasks with DQ checks, parallel and sequential).png	1.0.3	Yes		
Example_04.xml	1.0.8	Yes		
Example_05 (extending example 4 with an approval step).png	1.0.2	Yes		
Example_05.xml	1.0.8	Yes		

- Select one of the xml files, i.e. Example_01.xml click on it and copy the content into your clipboard.
- Now navigate to Catalog/Contributions/project:/StepWorkflow and click on it.
- Select one of the xml files, i.e. Workflow_01.xml and click on it to enable the editing mode.
- Copy the content of your clipboard and paste it into the browser window.
- Click on update.
- Make the necessary adjustments in the file: StepWorkflowMap.xml. (It is recommended to use Workflow_01.xml for the Example_01.xml and so on.)
- Define the trigger in the perspective "Business process management" in your MDM-Product 360 Desktop client.
- Finished!

3.3.6.2 Example 1 (1 Task)

3.3.6.3 Example 2 (2 Tasks)

3.3.6.4 Example 3 (2 Tasks with DQ checks)

3.3.6.5 Example 4 (5 Tasks with DQ checks, parallel and sequential)

3.3.6.6 Example 5 (extending example 4 with an approval step)

3.3.6.7 Example 6 (2 tasks (incl. 1 supplier task) + modifying of fields)

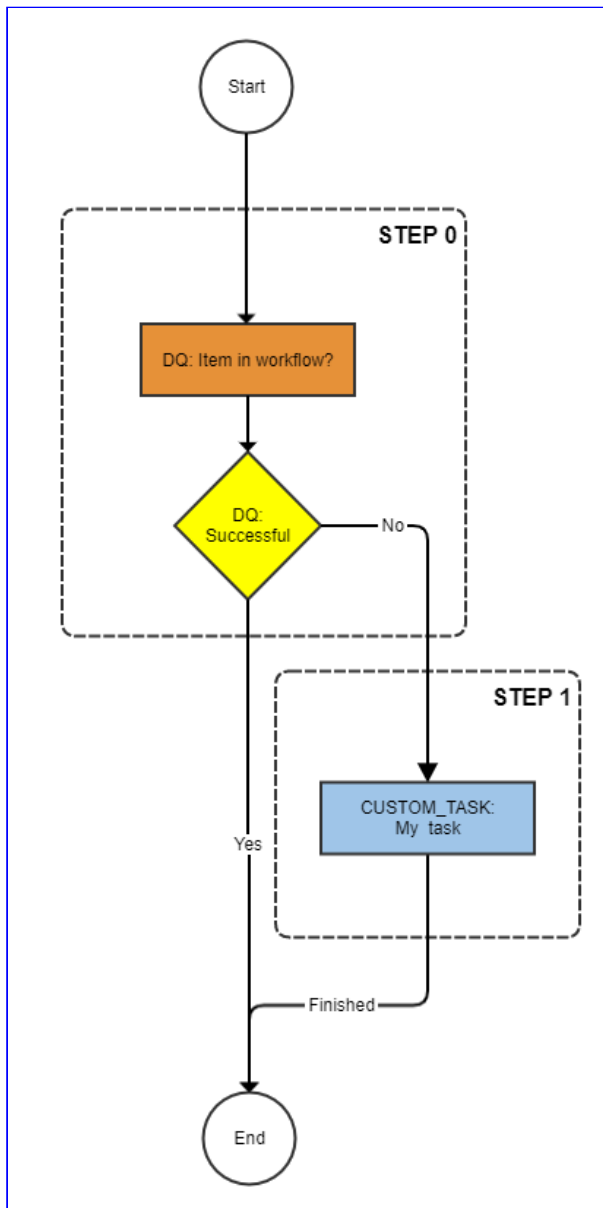
3.3.6.8 Example 7 (2 Tasks + modifying of fields + approval step)

3.3.6.9 Example 8 (2 Tasks modifying of fields approval step merge)

3.3.6.10 Example 9 (Classic task)

In this example we are creating a classic (standard) task. The difference between a workflow and a classic task is when finishing a classic task all items (products or variants) within this task will be finished at once. It is highly recommended to use this approach only with report based events like export completed, import started etc., because there will be always a new task created. Independent whether the same task with the same name for the entire catalog already exists.

Diagram for example 9



Explanation of the steps example 9

Step	Description
0	Check whether item is already in this workflow.
1	Put the item into the task "My task".

Product 360 prerequisites for example 9

Product 360 entity	value
Usergroup	Retail Team
Usergroup	Sales Team
Usergroup	Managers
Supplier	Tools Manufacturer
Supplier	Electronic Equipment Supply
Supplier	Fashion Unlimited
Ui Template	Item approve UI

Mapping file ManufacturerTeams.xml

This xml file defines the mapping between manufacturer and the default usergroup for the workflow task. If the manufacturer name is not found in the xml file, the task will be automatically assigned to the group **Managers**. Keep in mind this is only an example of a customized bpel.

ManufacturerTeams.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<teams>
  <team name="Retail Team">
    <manufacturer>Fashion Unlimited</manufacturer>
  </team>
  <team name="Sales Team">
    <manufacturer>Tools Manufacturer</manufacturer>
    <manufacturer>Electronic Equipment Supply</manufacturer>
  </team>
</teams>
```

StepWorkflow.xml file for example 9

Example 1

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 9</label>
```

```

<identifier>Workflow_09</identifier>
<version>1.0</version>
<step>
  <id>0</id>
  <entity>Article</entity>
  <enterStatus>Never</enterStatus>
  <batchSize>500</batchSize>
  <executeDq>Always</executeDq>
  <dqService>ItemsInWorkflowTasks-Process</dqService>
  <dqFailStep>STEP:1</dqFailStep>
</step>
<step>
  <id>1</id>
  <entity>Article</entity>
  <workflowStatus>My task</workflowStatus>
  <description>My workflow task</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>Never</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Never</executeDq>
  <getField>Article.ManufacturerName</getField>
  <nextStep>CustomTaskSample-Workflow</nextStep>
</step>
</workflow>
</imp:payload>

```

Detailed explanation of the steps

Step 0

This steps checks only whether the item is already in this workflow. If not the executed next step is 1. If yes the workflow will end because there is no next step defined.

Key	Value	Description
id	0	Identifier of the step.
entity	Article	Entity container for this workflow.
enterStatus	Never	The circumstances when an item should enter a status (task). In this case never!
batchSize	500	Batch size to use with trigger batching.
dqService	ItemsInWorkflowTasks-Process	This DQ service is an additional bpel inside the StepWorkflow which checks whether the item is already inside this workflow or not.

Key	Value	Description
dqFailStep	STEP:1	The item will be moved to the step with the identifier 1 if the above dq fails.

Step 1

This steps checks only whether the item is already in this workflow. If not the executed next step is 1. If yes the workflow will end because there is no next step defined.

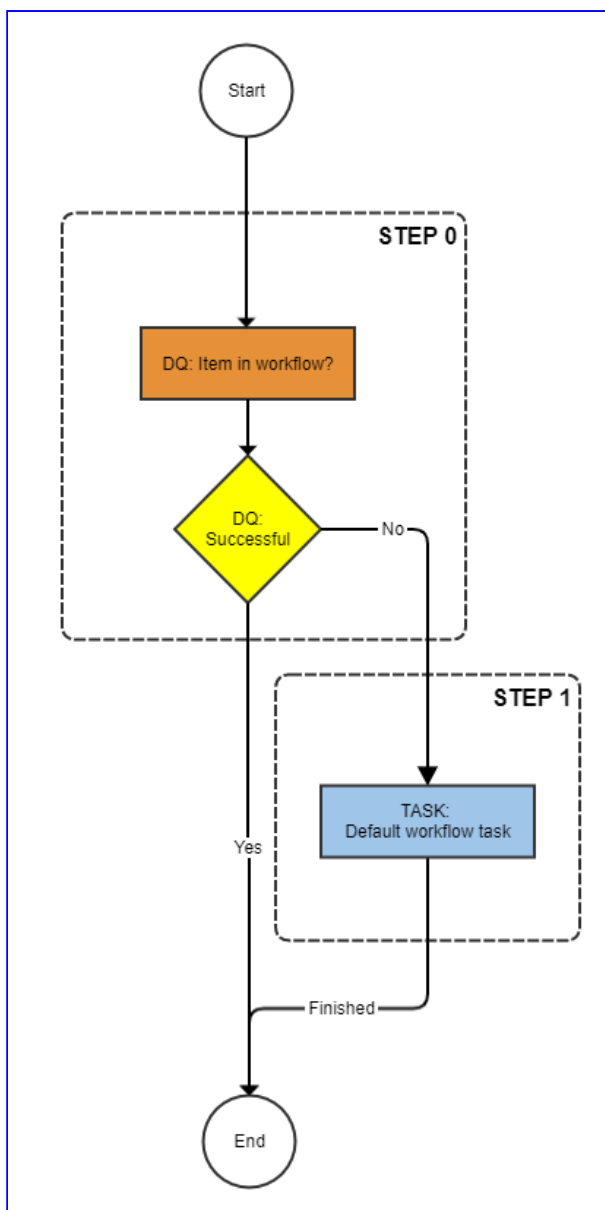
Key	Value	Description
id	1	Identifier of the step.
entity	Article	Entity container for this workflow.
workflow Status	My task	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
descripti on	My workflow task	Description of the status.
workflow ServiceE ndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.
enterStat us	Never	The circumstances when an item should enter a status (task). In this case never, because the task will be enetered in the custom bpel!
batchSiz e	500	Batch size to use with trigger batching.
userType	userGroup	Type of the default assignee of this task. In this case it will be a usergroup.
userNam e	Standardusers	Name/Identifier of the Default assignee of the task. ==> The item will be assigned to the usergroup "Standardusers".
uiTempla te	Item approve UI	This is the default UI of the task.
executeD q	Never	The circumstances when a DQ should be run. In this case never!
getField	Article.Manufactu rerName	List of any fields directly on the entity that will be required. These entries must be fully qualified.
nextStep	CustomTaskSam ple-Workflow	Endpoint of the bpel CustomTaskSample.

3.3.7 Defaults

The StepWorkflow package includes 3 default workflows. 1 workflow for items, 1 for products and an additional 1 for variants. These 3 workflows are only simple examples of defaults. Such a default workflow will be triggered if the trigger you defined in your Product 360 Desktop client does not match to an entry in the StepWorkflowMap.xml file.

3.3.7.1 DefaultArticle

Diagram for default article



Explanation of the steps default article

Step	Description
0	Check whether item is already in this workflow.
1	Put the item into the task "Default workflow task".

Product 360 prerequisites for default article

Product 360 entity	value
Usergroup	Standardusers

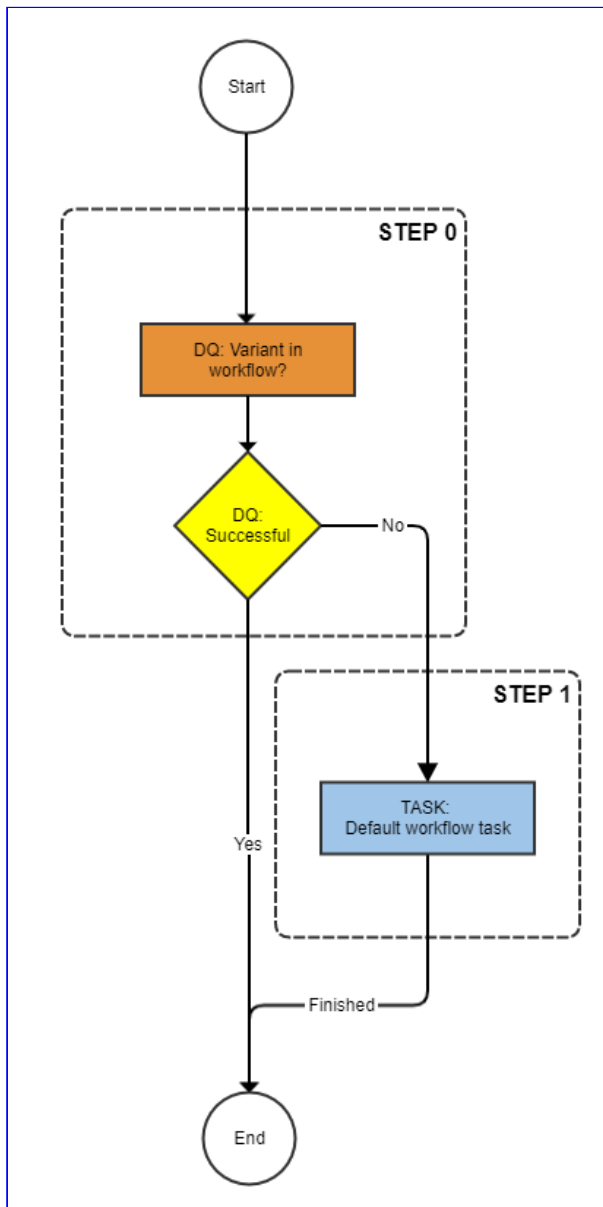
StepWorkflow.xml file for defaultArticle.xml

Example 1

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Item Workflow</label>
    <identifier>Default_Workflow_Article</identifier>
    <version>1.0</version>
    <step>
      <id>1</id>
      <entity>Article</entity>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:2</dqFailStep>
    </step>
    <step>
      <id>2</id>
      <entity>Article</entity>
      <workflowStatus>Default workflow task</workflowStatus>
      <description>Default workflow task</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
      <enterStatus>Always</enterStatus>
      <batchSize>500</batchSize>
      <userType>userGroup</userType>
      <userName>Standardusers</userName>
      <uiTemplate></uiTemplate>
      <executeDq>Never</executeDq>
    </step>
  </workflow>
</imp:payload>
```

3.3.7.2 DefaultVariant

Diagram for default variant



Explanation of the steps default variant

Step	Description
0	Check whether item is already in this workflow.
1	Put the item into the task "Default workflow task".

Product 360 prerequisites for default variant

Product 360 entity	value
Usergroup	Standardusers

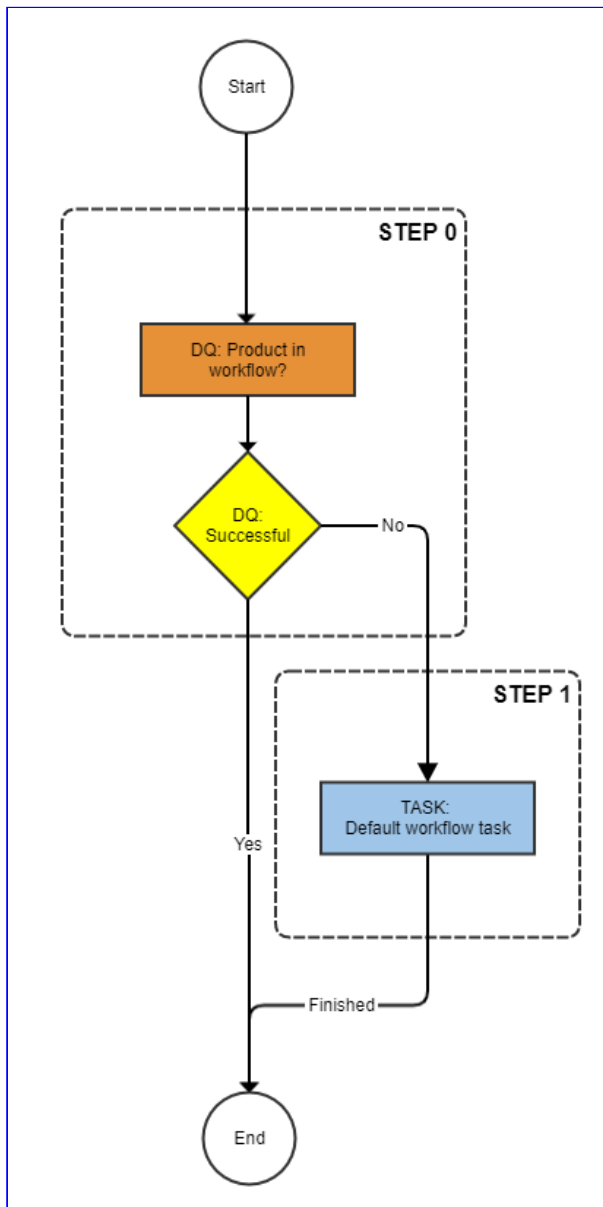
StepWorkflow.xml file for defaultVariant.xml

Example 1

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Variant Workflow</label>
    <identifier>Default_Workflow_Variant</identifier>
    <version>1.0</version>
    <step>
      <id>1</id>
      <entity>Variant</entity>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:2</dqFailStep>
    </step>
    <step>
      <id>2</id>
      <entity>Variant</entity>
      <workflowStatus>Default workflow task</workflowStatus>
      <description>Default workflow task</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
      <enterStatus>Always</enterStatus>
      <batchSize>500</batchSize>
      <userType>userGroup</userType>
      <userName>Standardusers</userName>
      <uiTemplate></uiTemplate>
      <executeDq>Never</executeDq>
    </step>
  </workflow>
</imp:payload>
```

3.3.7.3 DefaultProduct2G

Diagram for default product2g



Explanation of the steps default product2g

Step	Description
0	Check whether item is already in this workflow.
1	Put the item into the task "Default workflow task".

Product 360 prerequisites for default product2g

Product 360 entity	value
Usergroup	Standardusers

StepWorkflow.xml file for defaultProduct2G.xml

Example 1

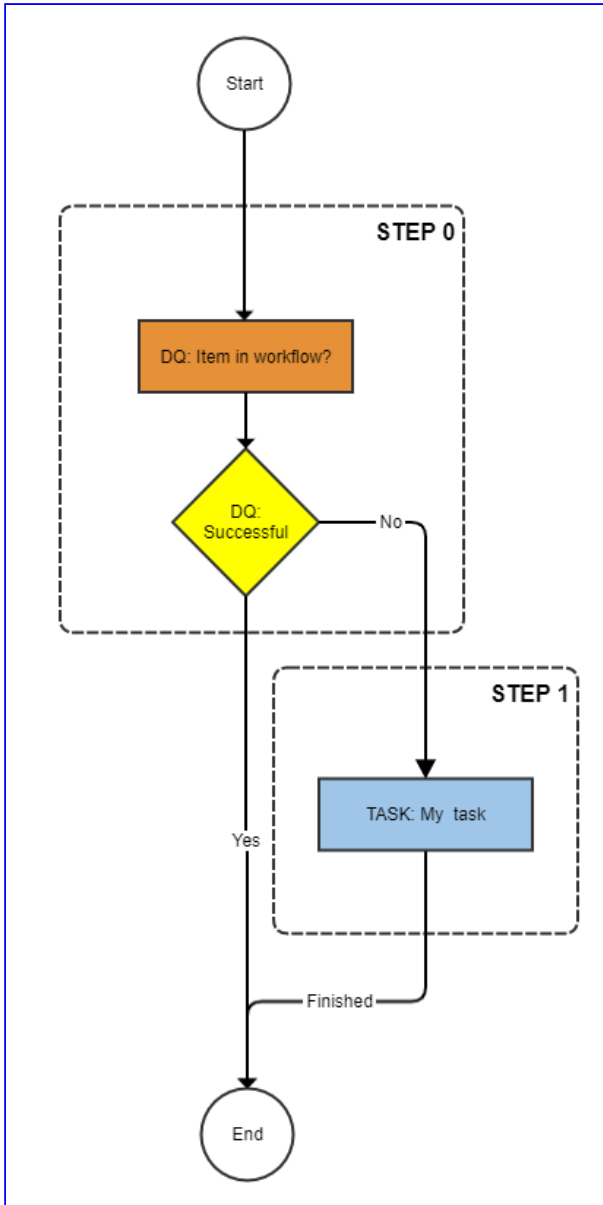
```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Product Workflow</label>
    <identifier>Default_Workflow_Product2G</identifier>
    <version>1.0</version>
    <step>
      <id>1</id>
      <entity>Product2G</entity>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:2</dqFailStep>
    </step>
    <step>
      <id>2</id>
      <entity>Product2G</entity>
      <workflowStatus>Default workflow task</workflowStatus>
      <description>Default workflow task</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
      <enterStatus>Always</enterStatus>
      <batchSize>500</batchSize>
      <userType>userGroup</userType>
      <userName>Standardusers</userName>
      <uiTemplate></uiTemplate>
      <executeDq>Never</executeDq>
    </step>
  </workflow>
</imp:payload>
```

3.3.8 Example 1 (1 Task)

The first workflow only puts items into a workflow task called “My first task” For this workflow task we need 2 steps. The first step checks whether the affected item is already in this workflow, if not it will be put into the task. The first step is mandatory and will be very useful in subsequent examples. With this approach you

can ensure that the item is not in the same workflow more than one time. This is very important for typical approval workflows.

3.3.8.1 Diagram for example 1



3.3.8.2 Explanation of the steps example 1

Step	Description
0	Check whether item is already in this workflow.
1	Put the item into the task "My first task".

3.3.8.3 Product 360 prerequisites for example 1

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI



Q Info

Standarduser is the Identifier of the Usergroup!

3.3.8.4 StepWorkflow.xml file for example 1

Example 1

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 1</label>
    <identifier>Workflow_01</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:1</dqFailStep>
    </step>
    <step>
      <id>1</id>
      <entity>Article</entity>
      <workflowStatus>My first task</workflowStatus>
      <description>My first workflow task</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
      <enterStatus>Always</enterStatus>
      <userType>userGroup</userType>
      <userName>Standardusers</userName>
      <uiTemplate>Item approve UI</uiTemplate>
      <executeDq>Never</executeDq>
    </step>
  </workflow>
</imp:payload>
</payload>
```



```
</workflow>
</imp:payload>
```

3.3.8.5 Detailed explanation of the steps

Step 0

This steps checks only whether the item is already in this workflow. If not the executed next step is 1. If yes the workflow will end because there is no next step defined.

Key	Value	Description
id	0	Identifier of the step.
entity	Article	Entity container for this workflow.
enterStatus	Never	The circumstances when an item should enter a status (task). In this case never!
batchSize	500	Batch size to use with trigger batching.
dqService	ItemsInWorkflowTasks-Process	This DQ service is an additional bpel inside the StepWorkflow which checks whether the item is already inside this workflow or not.
dqFailStep	STEP:1	The item will be moved to the step with the identifier 1 if the above dq fails.

Step 1

This steps checks only whether the item is already in this workflow. If not the executed next step is 1. If yes the workflow will end because there is no next step defined.

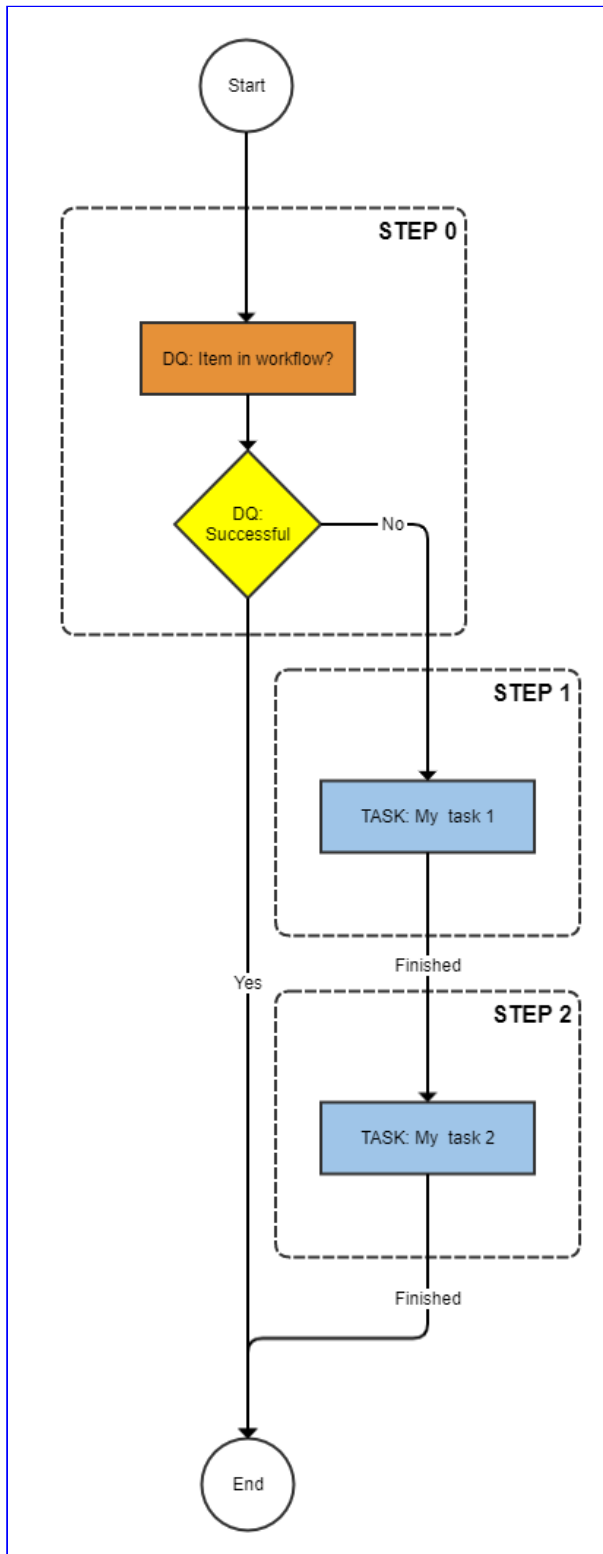
Key	Value	Description
id	1	Identifier of the step.
entity	Article	Entity container for this workflow.
workflow Status	My first task	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	My first workflow task	Description of the status.
workflow ServiceEndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.

Key	Value	Description
enterStatus	Always	The circumstances when an item should enter a status (task). In this case always!
userType	userGroup	Type of the default assignee of this task. In this case it will be a usergroup.
userName	Standardusers	Name/Identifier of the Default assignee of the task. ==> The item will be assigned to the usergroup "Standardusers".
uiTemplate	Item approve UI	This is the default UI of the task.
executeDq	Never	The circumstances when a DQ should be run. In this case never!

3.3.9 Example 2 (2 Tasks)

The next example extends the first example with a further task after the first task.

3.3.9.1 Diagram for example 2



3.3.9.2 Product 360 prerequisites for example 2

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI

3.3.9.3 Explanation of the steps example 2

Step	Description
0	Check whether item is already in this workflow.
1	Put the item into the task "My task 1".
2	Put the item into the task "My task 2".

3.3.9.4 StepWorkflow.xml file for example 2

Example 2

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 2</label>
    <identifier>Workflow_02</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:1</dqFailStep>
    </step>
    <step>
      <id>1</id>
      <entity>Article</entity>
      <workflowStatus>My task 1</workflowStatus>
      <description>My task 1</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
      <enterStatus>Always</enterStatus>
      <batchSize>500</batchSize>
    </step>
  </workflow>
</imp:payload>
</imp:payload>
```

```

<userType>userGroup</userType>
<userName>Standardusers</userName>
<uiTemplate>Item approve UI</uiTemplate>
<executeDq>Never</executeDq>
<nextStep>STEP:2</nextStep>
</step>
<step>
  <id>2</id>
  <entity>Article</entity>
  <workflowStatus>My task 2</workflowStatus>
  <description>My task 2</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>Always</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Never</executeDq>
</step>
</workflow>
</imp:payload>

```

3.3.9.5 Detailed explanation of the steps

Only the differences to the previous example will be explained.

Step 0

Step 0 is an exact copy of the step 0 of the example 1.

Step 1

Step 1 is nearly the same as the step 1 of the example above. Only the differences are shown in the next table.

Key	Value	Description
...	...	Same keys and values than in example 1.
workflow Status	My task 1	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	My task 1	Description of the status.
nextStep	STEP:2	Identifier of the next step, The item will be moved to the next step, when this step is finished.

Step 2

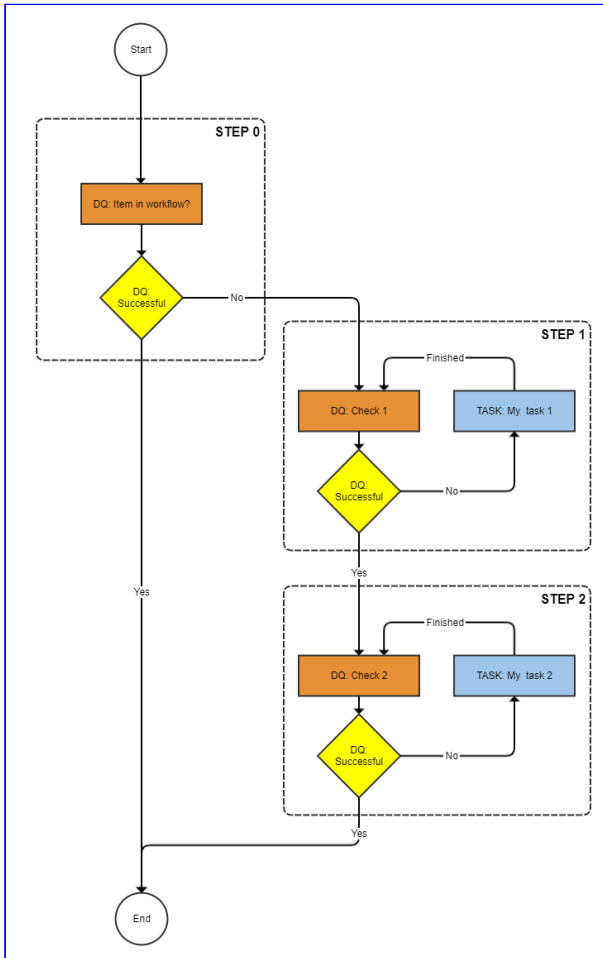
Step 2 is mostly a copy of step 1.

Key	Value	Description
...	...	Same keys and values than step 1.
id	2	Identifier of the step.
workflow Status	My task 2	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
descripti on	My task 2	Description of the status.
nextStep		This step contains no next step, because the workflow have to end at this point.

3.3.10 Example 3 (2 Tasks with DQ checks)

The next example extends the previous example so that the tasks will only be created when the DQ check fails.

3.3.10.1 Diagram for example 3



3.3.10.2 Product 360 prerequisites for example 3

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI
DQ Channel	Channel 1
DQ Channel	Channel 2

3.3.10.3 Explanation of the steps example 3

Step	Description
0	Check whether item is already in this workflow.

Step	Description
1	Run DQ check "Channel 1" if fail ==>put the item into the task "My task 1"
2	Run DQ check "Channel 2" if fail ==>put the item into the task "My task 2"

3.3.10.4 StepWorkflow.xml file for example 3

Example 3

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 3</label>
    <identifier>Workflow_03</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:1</dqFailStep>
    </step>
    <step>
      <id>1</id>
      <entity>Article</entity>
      <workflowStatus>My task 1</workflowStatus>
      <description>My task 1</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
      <enterStatus>OnDqResults</enterStatus>
      <batchSize>500</batchSize>
      <userType>userGroup</userType>
      <userName>Standardusers</userName>
      <uiTemplate>Item approve UI</uiTemplate>
      <executeDq>Always</executeDq>
      <dqService>ExecuteDqBatch-Process</dqService>
      <dqChannel>Channel 1</dqChannel>
      <nextStep>STEP:2</nextStep>
    </step>
    <step>
      <id>2</id>
      <entity>Article</entity>
      <workflowStatus>My task 2</workflowStatus>
      <description>My task 2</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
```



```

    <enterStatus>OnDqResults</enterStatus>
    <batchSize>500</batchSize>
    <userType>userGroup</userType>
    <userName>Standardusers</userName>
    <uiTemplate>Item approve UI</uiTemplate>
    <executeDq>Always</executeDq>
    <dqService>ExecuteDqBatch-Process</dqService>
    <dqChannel>Channel 2</dqChannel>
  </step>
</workflow>
</imp:payload>

```

3.3.10.5 Detailed explanation of the steps

Only the differences to the previous example will be explained.

Step 0

Step 0 is an exact copy of the step 0 of the example 2.

Step 1

Step 1 is nearly the same as the step 1 of example 2. Only the differences are shown in the next table.

Key	Value	Description
...	...	Same keys and values than in example 2.
enterStatus	OnDqResults	The circumstances when an item should enter a status (task). In this case only when DQ check will fail!
executeDq	Always	The circumstances when a DQ should be run. In this case always!
dqService	ExecuteDqBatch-Process	The service name of the process that will be handle the processing of the dq.
dqChannel	Channel 1	Name of the DQ channel which will be executed.

Step 2

Step 2 is nearly the same as the step 1 of the example above.

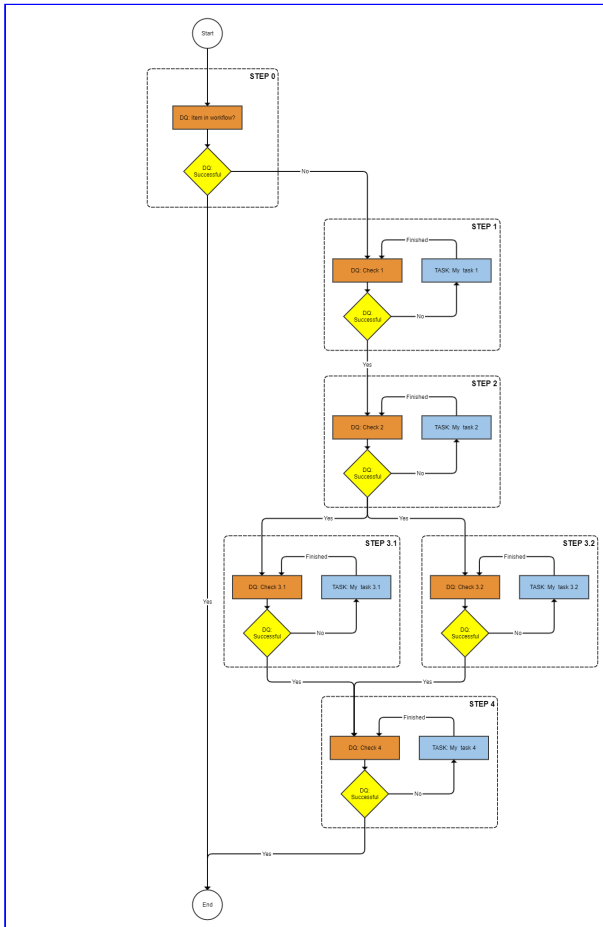
Only the differences are shown in the next table.

Key	Value	Description
...	...	Same keys and values than in example 2.
enterStatus	OnDqResults	The circumstances when an item should enter a status (task). In this case only when DQ check will fail!
executeDq	Always	The circumstances when a DQ should be run. In this case always!
dqService	ExecuteDqBatch-Process	The service name of the process that will be handle the processing of the DQ.
dqChannel	Channel 2	Name of the DQ channel which will be executed.

3.3.11 Example 4 (5 Tasks with DQ checks, parallel and sequential)

Example 4 starts with 2 sequential tasks, after that there are 2 tasks which are running in parallel and after finishing those 2 parallel tasks the last task will start.

3.3.11.1 Diagram for example 4



3.3.11.2 Product 360 prerequisites for example 4

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI
DQ Channel	Channel 1
DQ Channel	Channel 2
DQ Channel	Channel 3.1
DQ Channel	Channel 3.2
DQ Channel	Channel 4

3.3.11.3 Explanation of the steps example 4

Step	Description
0	Check whether item is already in this workflow.
1	Run DQ check "Channel 1" if fail ==>put the item into the task "My task 1"
2	Run DQ check "Channel 2" if fail ==>put the item into the task "My task 2"
3.1	Run DQ check "Channel 3.1" if fail ==>put the item into the task "My task 3.1"
3.2	Run DQ check "Channel 3.2" if fail ==>put the item into the task "My task 3.2"
4	Run DQ check "Channel 4" if fail ==>put the item into the task "My task 4"

3.3.11.4 StepWorkflow.xml file for example 4

Example 4

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 4</label>
    <identifier>Workflow_04</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:1</dqFailStep>
    </step>
    <step>
      <id>1</id>
      <entity>Article</entity>
      <workflowStatus>My task 1</workflowStatus>
      <description>My task 1</description>
      <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
      <enterStatus>OnDqResults</enterStatus>
      <batchSize>500</batchSize>
      <userType>userGroup</userType>
      <userName>Standardusers</userName>
      <uiTemplate>Item approve UI</uiTemplate>
    </step>
  </workflow>
</imp:payload>
```

```

    <executeDq>Always</executeDq>
    <dqService>ExecuteDqBatch-Process</dqService>
    <dqChannel>Channel 1</dqChannel>
    <nextStep>STEP:2</nextStep>
</step>
<step>
  <id>2</id>
  <entity>Article</entity>
  <workflowStatus>My task 2</workflowStatus>
  <description>My task 2</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>OnDqResults</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Always</executeDq>
  <dqService>ExecuteDqBatch-Process</dqService>
  <dqChannel>Channel 2</dqChannel>
  <nextStep>StepWorkflow_ParallelTasks-Workflow</nextStep>
  <parallelStep>3.1</parallelStep>
  <parallelStep>3.2</parallelStep>
  <parallelNextStep>4</parallelNextStep>
</step>
<step>
  <id>3.1</id>
  <entity>Article</entity>
  <workflowStatus>My task 3.1</workflowStatus>
  <description>My task 3.1</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>OnDqResults</enterStatus>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Always</executeDq>
  <dqService>ExecuteDqBatch-Process</dqService>
  <dqChannel>Channel 3.1</dqChannel>
  <dqFailTrigger>StepWorkflow_ParallelTasks-Finish-1</dqFailTrigger>
  <nextTrigger>StepWorkflow_ParallelTasks-Finish-1</nextTrigger>
</step>
<step>
  <id>3.2</id>
  <entity>Article</entity>
  <workflowStatus>My task 3.2</workflowStatus>
  <description>My task 3.2</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>OnDqResults</enterStatus>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Always</executeDq>
  <dqService>ExecuteDqBatch-Process</dqService>

```

```

<dqChannel>Channel 3.2</dqChannel>
<dqFailTrigger>StepWorkflow_ParallelTasks-Finish-2</dqFailTrigger>
<nextTrigger>StepWorkflow_ParallelTasks-Finish-2</nextTrigger>
</step>
<step>
  <id>4</id>
  <entity>Article</entity>
  <workflowStatus>My task 4</workflowStatus>
  <description>My task 4</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>OnDqResults</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Always</executeDq>
  <dqService>ExecuteDqBatch-Process</dqService>
  <dqChannel>Channel 4</dqChannel>
</step>
</workflow>
</imp:payload>

```

3.3.11.5 Detailed explanation of the steps

Only the differences to the previous example will be explained.

Step 0

Step 0 is an exact copy of the step 0 of example 3.

Step 1

Step 1 is an exact copy of the step 1 of example above 3.

Step 2

Step 2 is nearly the same as the step 2 of the example above. Only the differences are shown in the next table.

Key	Value	Description
...	...	Same keys and values than in example 3.
nextStep	StepWorkflow_ParallelTasks-Workflow	StepWorkflow_ParallelTasks-Workflow defines that the next steps run in parallel

Key	Value	Description
parallel Step	3.1	Identifier of the first parallel step, the item will be moved to this parallel step. The maximum number of parallel steps is 6.
parallel Step	3.2	Identifier of the second parallel step, the item will be moved also to this parallel step.
parallel NextStep	4	The item will be moved to this step when all parallel steps are finished.

Step 3.1

Key	Value	Description
id	3.1	Identifier of the step.
entity	Article	Entity container for this workflow.
workflowStatus	My task 3.1	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	My task 3.1	Description of the status.
workflowServiceEndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.
enterStatus	OnDqResults	The circumstances when an item should enter a status (task). In this case only when DQ check will fail!
userType	userGroup	Type of the default assignee of this task. In this case it will be a usergroup.
userName	Standardusers	Name/Identifier of the Default assignee of the task. ==> The item will be assigned to the usergroup "Standardusers".
uiTemplate	Item approve UI	This is the default UI of the task.
executeDq	Always	The circumstances when a DQ should be run. In this case always!
dqService	ExecuteDqBatch-Process	The service name of the process that will be handle the processing of the DQ.
dqChannel	Channel 3.1	Name of the DQ channel which will be executed.

Key	Value	Description
dqFailTrigger	StepWorkflow_ParallelTasks-Finish-1	This is the service name of the partner link of the parallel tasks. It will be called if DQ fails to ensure that the item will remain in the status (task).
nextTrigger	StepWorkflow_ParallelTasks-Finish-1	This is another service name of the partner link of the parallel tasks. It will be called if the item will leave this step.

Step 3.2

Step 3.2 is nearly the same as the step 3.1. Only the differences are shown in the next table.

Key	Value	Description
id	3.1	Identifier of the step.
workflowStatus	My task 3.2	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	My task 3.2	Description of the status.
dqChannel	Channel 3.2	Name of the DQ channel which will be executed.
dqFailTrigger	StepWorkflow_ParallelTasks-Finish-2	This is the service name of the partner link of the parallel tasks. It will be called if DQ fails to ensure that the item will remain in the status (task).
nextTrigger	StepWorkflow_ParallelTasks-Finish-2	This is another service name of the partner link of the parallel tasks. It will be called if the item will leave this step.

Step 4

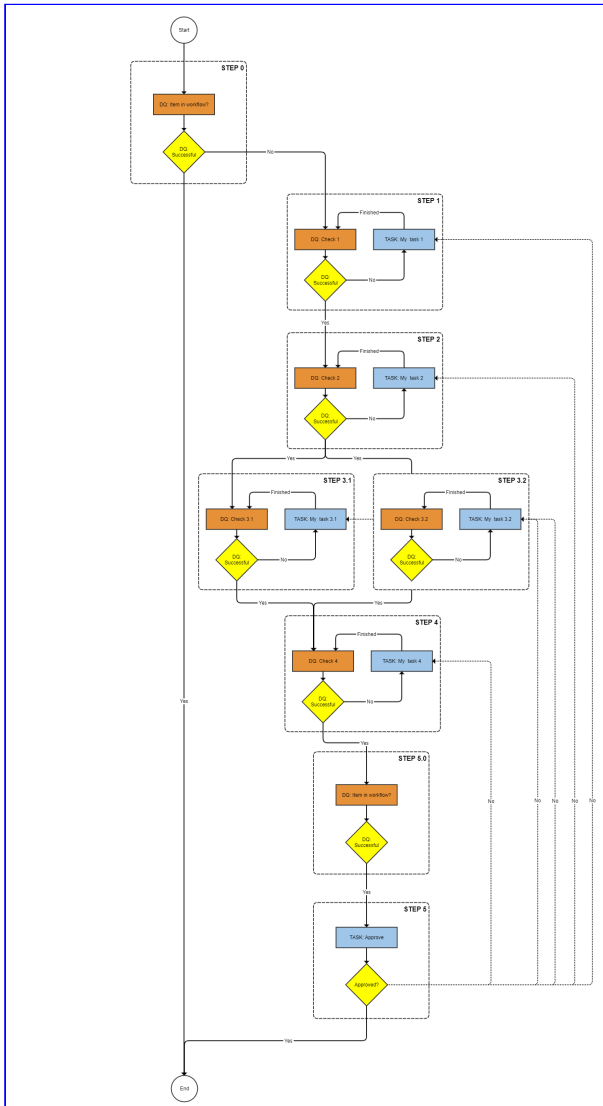
Key	Value	Description
id	4	Identifier of the step.
entity	Article	Entity container for this workflow.
workflowStatus	My task 4	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	My task 4	Description of the status.
workflowServiceEndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.
enterStatus	OnDqResults	The circumstances when an item should enter a status (task). In this case only when DQ check will fail!

Key	Value	Description
batchSize	500	Batch size to use with trigger batching.
userType	userGroup	Type of the default assignee of this task. In this case it will be a usergroup.
userName	Standardusers	Name/Identifier of the Default assignee of the task. ==> The item will be assigned to the usergroup "Standardusers".
uiTemplate	Item approve UI	This is the default UI of the task.
executeDq	Always	The circumstances when a DQ should be run. In this case always!
dqService	ExecuteDqBatch-Process	The service name of the process that will be handle the processing of the DQ.
dqChannel	Channel 4	Name of the DQ channel which will be executed.

3.3.12 Example 5 (extending example 4 with an approval step)

The next example extends the previous example with an approval step at the end of the workflow, so the approver can reject the item to all of the previous steps.

3.3.12.1 Diagram for example 5



3.3.12.2 Product 360 prerequisites for example 5

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI
DQ Channel	Channel 1
DQ Channel	Channel 2
DQ Channel	Channel 3.1

Product 360 entity	value
DQ Channel	Channel 3.2
DQ Channel	Channel 4

3.3.12.3 Explanation of the steps example 5

Step	Description
0	Check whether item is already in this workflow
1	Run DQ check "Channel 1" if fail ==>put the item into the task "My task 1"
2	Run DQ check "Channel 2" if fail ==>put the item into the task "My task 2"
3.1	Run DQ check "Channel 3.1" if fail ==>put the item into the task "My task 3.1"
3.2	Run DQ check "Channel 3.2" if fail ==>put the item into the task "My task 3.2"
4	Run DQ check "Channel 4" if fail ==>put the item into the task "My task 4"
5.0	Check whether item is already in this workflow.
5	Approve task with rejection options to all previous tasks

3.3.12.4 StepWorkflow.xml file for example 5

Example 5

```
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 5</label>
    <identifier>Workflow_05</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:1</dqFailStep>
    </step>
    <step>
      <id>1</id>
```

```

<entity>Article</entity>
<workflowStatus>My task 1</workflowStatus>
<description>My task 1</description>
<workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
<enterStatus>OnDqResults</enterStatus>
<batchSize>500</batchSize>
<userType>userGroup</userType>
<userName>Standardusers</userName>
<uiTemplate>Item approve UI</uiTemplate>
<executeDq>Always</executeDq>
<dqService>ExecuteDqBatch-Process</dqService>
<dqChannel>Channel 1</dqChannel>
<nextStep>STEP:2</nextStep>
</step>
<step>
  <id>2</id>
  <entity>Article</entity>
  <workflowStatus>My task 2</workflowStatus>
  <description>My task 2</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>OnDqResults</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Always</executeDq>
  <dqService>ExecuteDqBatch-Process</dqService>
  <dqChannel>Channel 2</dqChannel>
  <nextStep>StepWorkflow_ParallelTasks-Workflow</nextStep>
  <parallelStep>3.1</parallelStep>
  <parallelStep>3.2</parallelStep>
  <parallelNextStep>4</parallelNextStep>
</step>
<step>
  <id>3.1</id>
  <entity>Article</entity>
  <workflowStatus>My task 3.1</workflowStatus>
  <description>My task 3.1</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>OnDqResults</enterStatus>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Always</executeDq>
  <dqService>ExecuteDqBatch-Process</dqService>
  <dqChannel>Channel 3.1</dqChannel>
  <dqFailTrigger>StepWorkflow_ParallelTasks-Finish-1</dqFailTrigger>
  <nextTrigger>StepWorkflow_ParallelTasks-Finish-1</nextTrigger>
</step>
<step>
  <id>3.2</id>
  <entity>Article</entity>

```

```

<workflowStatus>My task 3.2</workflowStatus>
<description>My task 3.2</description>
<workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
<enterStatus>OnDqResults</enterStatus>
<userType>userGroup</userType>
<userName>Standardusers</userName>
<uiTemplate>Item approve UI</uiTemplate>
<executeDq>Always</executeDq>
<dqService>ExecuteDqBatch-Process</dqService>
<dqChannel>Channel 3.2</dqChannel>
<dqFailTrigger>StepWorkflow_ParallelTasks-Finish-2</dqFailTrigger>
<nextTrigger>StepWorkflow_ParallelTasks-Finish-2</nextTrigger>
</step>
<step>
  <id>4</id>
  <entity>Article</entity>
  <workflowStatus>My task 4</workflowStatus>
  <description>My task 4</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>OnDqResults</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Always</executeDq>
  <dqService>ExecuteDqBatch-Process</dqService>
  <dqChannel>Channel 4</dqChannel>
  <nextStep>STEP:5.0</nextStep>
</step>
<step>
  <id>5.0</id>
  <entity>Article</entity>
  <enterStatus>Never</enterStatus>
  <batchSize>500</batchSize>
  <executeDq>Always</executeDq>
  <dqService>ItemsInWorkflowTasks-Process</dqService>
  <dqFailStep>STEP:5</dqFailStep>
</step>
<step>
  <id>5</id>
  <entity>Article</entity>
  <workflowStatus>Approve</workflowStatus>
  <description>Approval</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>Always</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Never</executeDq>
  <rejectTrigger>StepWorkflow_ParallelTasks-Reject</rejectTrigger>
  <singleChoice>>false</singleChoice>

```

```

<rejectDecision>
  <id>p360.bpm.reject.STEP:1</id>
  <label>Reject to task 1</label>
</rejectDecision>
<rejectDecision>
  <id>p360.bpm.reject.STEP:2</id>
  <label>Reject to task 2</label>
</rejectDecision>
<rejectDecision>
  <id>p360.bpm.reject.TRIGGER:3.1</id>
  <label>Reject to task 3.1</label>
</rejectDecision>
<rejectDecision>
  <id>p360.bpm.reject.TRIGGER:3.2</id>
  <label>Reject to task 3.2</label>
</rejectDecision>
<rejectDecision>
  <id>p360.bpm.reject.STEP:4</id>
  <label>Reject to task 4</label>
</rejectDecision>
</step>
</workflow>
</imp:payload>

```

3.3.12.5 Detailed explanation of the steps

Only the differences to the previous example will be explained.

Step 0

Step 0 is an exact copy of the step 0 of the example 4.

Step 1

Step 1 is an exact copy of the step 1 of the example 4.

Step 2

Step 2 is an exact copy of the step 2 of the example 4.

Step 3.1

Step 3.1 is an exact copy of the step 3.1 of the example 4.

Step 3.2

Step 3.2 is an exact copy of the step 3.2 of the example 4.

Step 4

Step 4 is nearly the same as the step 4 of the example 4. Only the differences are shown in the next table.

Key	Value	Description
...	...	Same keys and values than in the example 4.
nextStep	STEP:5.0	Identifier of the next step, The item will be moved to the next step, when this step is finished.

Step 5.0

This steps checks only whether the item is already in this workflow. If not the executed next step is 5. If yes the workflow will end to avoid that the item will go to the step 5 (Approve) until it is somewhere else in the workflow.

Key	Value	Description
id	5.0	Identifier of the step.
entity	Article	Entity container for this workflow.
enterStatus	Never	The circumstances when an item should enter a status (task). In this case never!
batchSize	500	Batch size to use with trigger batching.
dqService	ItemsInWorkflowTasks-Process	This DQ service is an additional bpel inside the StepWorkflow which checks whether the item is already inside this workflow or not.
dqFailStep	STEP:5	The item will be moved to the step with the identifier 5 if the above DQ fails.

Step 5

This steps checks only whether the item is already in this workflow. If not the executed next step is 5. If yes the workflow will end to avoid that the item will go to the step 5 (Approve) until it is somewhere else in the workflow.

Key	Value	Description
id	5	Identifier of the step.
entity	Article	Entity container for this workflow.

Key	Value	Description
workflowStatus	Approve	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	Approval	Description of the status.
workflowServiceEndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.
enterStatus	Always	The circumstances when an item should enter a status (task). In this case always!
batchSize	500	Batch size to use with trigger batching.
userType	userGroup	Type of the default assignee of this task. In this case it will be a usergroup.
userName	Standardusers	Name/Identifier of the Default assignee of the task. ==> The item will be assigned to the usergroup "Standardusers".
uiTemplate	Item approve UI	This is the default UI of the task.
executeDq	Never	The circumstances when a DQ should be run. In this case never!
rejectTrigger	StepWorkflow_ParallelTasks-Reject	Endpoint for the reject trigger of parallel states (tasks).
singleChoice	false	This is for Rejects only and determines if you can have more than one choice. ==> The workflow can be rejected to all steps in one decision.
<rejectDecision>id	p360.bpm.reject.STEP:1	Identifier of the 1 st reject step.
<rejectDecision>label	Reject to task 1	Label of the 1 st rejection. This label will show up in the UI.
<rejectDecision>id	p360.bpm.reject.STEP:2	Identifier of the 2 nd reject step.
<rejectDecision>label	Reject to task 2	Label of the 2 nd rejection. This label will show up in the UI.
<rejectDecision>id	p360.bpm.reject.TRIGGER:3.1	Identifier of the 3 rd reject step. (In this case you have to reject to a trigger!)
<rejectDecision>label	Reject to task 3.1	Label of the 3 rd rejection. This label will show up in the UI.

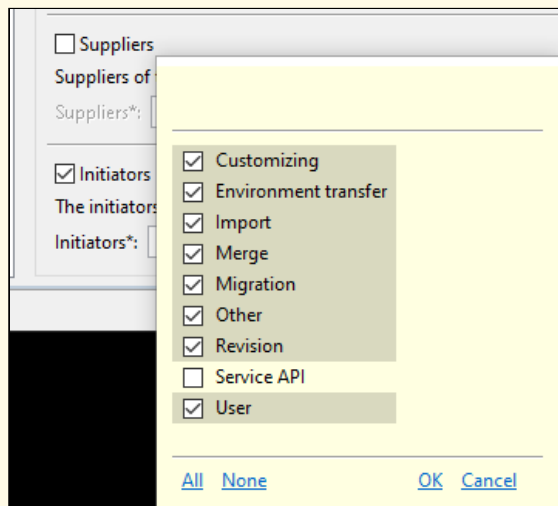
Key	Value	Description
<rejectDecision>id	p360.bpm.reject.TRIGGER:3.2	Identifier of the 4th reject step. (In this case you have to reject to a trigger!)
<rejectDecision>label	Reject to task 3.2	Label of the 4th rejection. This label will show up in the UI.
<rejectDecision>id	p360.bpm.reject.STEP:4	Identifier of the 5th reject step.
<rejectDecision>label	Reject to task 4	Label of the 5th rejection. This label will show up in the UI.

3.3.13 Example 6 (2 tasks (incl. 1 supplier task) + modifying of fields)

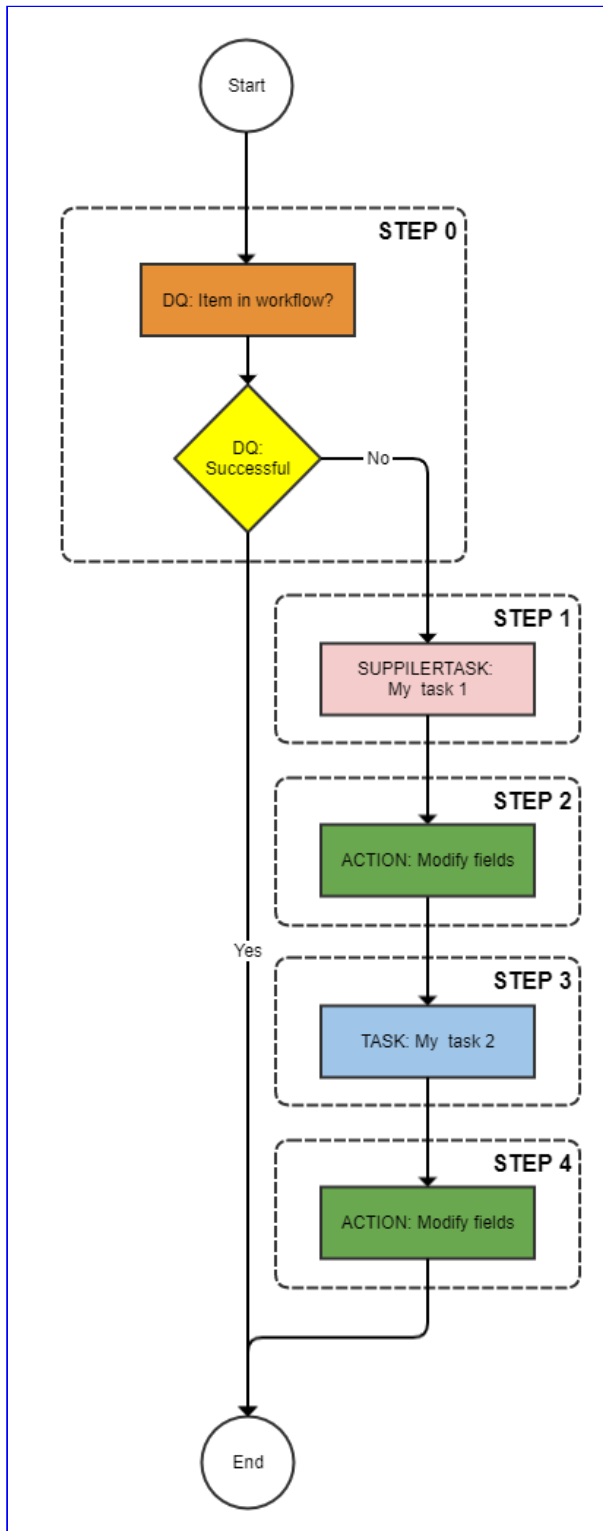
The next example extends the example 2 workflow with modifying of fields within the workflow after finishing the tasks,

Q Info

Please consider to define your trigger in Product 360 carefully, because if the initiator "Service API" is not excluded for this workflow will run in a loop!



3.3.13.1 Diagram for example 6



3.3.13.2 Product 360 prerequisites for example 6

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI

3.3.13.3 Explanation of the steps example 6

Step	Description
0	Check whether item is already in this workflow
1	Put the item into the supplier task "My task 1"
2	Modify the field Article.CurrentStatus to the value 200
3	Put the item into the task "My task 2"
4	Modify the field Article.CurrentStatus to the value 300

3.3.13.4 StepWorkflow.xml file for example 6

Example 6

```
<?xml version="1.0" encoding="UTF-8"?>
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 6</label>
    <identifier>Workflow_06</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
      <dqFailStep>STEP:1</dqFailStep>
      <getField>Article.MainSupplier->Party.Name</getField>
    </step>
    <step>
      <id>1</id>
      <entity>Article</entity>
```

```

    <workflowStatus>My task 1</workflowStatus>
    <description>My task 1</description>
    <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
    <enterStatus>Always</enterStatus>
    <batchSize>500</batchSize>
    <userType>supplier</userType>
    <uiTemplate>Item approve UI</uiTemplate>
    <executeDq>Never</executeDq>
    <getField>Article.CurrentStatus</getField>
    <nextStep>STEP:2</nextStep>
    <nextStep>STEP:3</nextStep>
  </step>
  <step>
    <id>2</id>
    <entity>Article</entity>
    <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
    <updateFieldValue>200</updateFieldValue>
  </step>
  <step>
    <id>3</id>
    <entity>Article</entity>
    <workflowStatus>My task 2</workflowStatus>
    <description>My task 2</description>
    <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
    <enterStatus>Always</enterStatus>
    <batchSize>500</batchSize>
    <userType>userGroup</userType>
    <userName>Standardusers</userName>
    <uiTemplate>Item approve UI</uiTemplate>
    <executeDq>Never</executeDq>
    <nextStep>STEP:4</nextStep>
  </step>
  <step>
    <id>4</id>
    <entity>Article</entity>
    <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
    <updateFieldValue>300</updateFieldValue>
  </step>
</workflow>
</imp:payload>

```

3.3.13.5 Detailed explanation of the steps

Step 0

Step 0 is an exact copy of the step 0 of the example 5 (incl. getField for supplier tasks).

Step 0

This steps checks only whether the item is already in this workflow. The difference to the previous step 0 steps is, that in this case the mandatory field "Article.MainSupplier->Party.Name " for supplier tasks is called. When you work with supplier tasks it is necessary that this getField command is part of the very first step of the StepWorkflow.

Key	Value	Description
id	0	Identifier of the step.
entity	Article	Entity container for this workflow.
enterStatus	Never	The circumstances when an item should enter a status (task). In this case never!
batchSize	500	Batch size to use with trigger batching.
dqService	ItemsInWorkflowTasks-Process	This DQ service is an additional bpel inside the StepWorkflow which checks whether the item is already inside this workflow or not.
dqFailStep	STEP:1	The item will be moved to the step with the identifier 1 if the above dq fails.
getField	Article.MainSupplier->Party.Name	List of any fields directly on the entity that will be required. These entries must be fully qualified.

Step 1

This step puts the item into a workflow task and when leaving this task the steps 2 and 3 are executed.

Key	Value	Description
id	1	Identifier of the step.
entity	Article	Entity container for this workflow.
workflow Status	My task 1	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	My task 1	Description of the status.
workflow ServiceEndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.

Key	Value	Description
enterStatus	Always	The circumstances when an item should enter a status (task). In this case always!
batchSize	500	Batch size to use with trigger batching.
userType	supplier	Type of the default assignee of this task. In this case it will be assigned to the supplier of the catalog.
uiTemplate	Item approve UI	This is the default UI of the task.
executeDq	Never	The circumstances when a DQ should be run. In this case never!
getField	Article.CurrentStatus	List of any fields directly on the entity that will be required. These entries must be fully qualified.
nextStep	STEP:2	Identifier of the next step, The item will be moved to the next step, when this step is finished.
nextStep	STEP:3	Identifier of the next step, The item will be moved to the next step, when this step is finished.

Step 2

Step 2 will manipulate data within the workflow. In this example we will change the field CurrentStatus of the entity Article.

Key	Value	Description
id	2	Same keys and values than in the example above.
entity	Article	Entity container for this workflow.
updateFieldDescriptor	Article.CurrentStatus	This field allows the step to update a field in Product 360. The field must be fully qualified.
updateFieldValue	200	This is the value for the update.

The next table shows you how to qualify field.

Key	Value
FieldDescriptor	Article.CurrentStatus

Key	Value
FieldDescriptor	Article.MainSupplier-&Party.Name
FieldDescriptor	Article.ManufacturerName
FieldDescriptor	ArticleLang.DescriptionShort(9,"Default Channel")

Step 3

This step puts the item again into a workflow task and when leaving this task the steps 4 is executed.

Key	Value	Description
id	3	Identifier of the step.
entity	Article	Entity container for this workflow.
workflow Status	My task 2	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	My task 2	Description of the status.
workflow ServiceEndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.
enterStatus	Always	The circumstances when an item should enter a status (task). In this case always!
batchSize	500	Batch size to use with trigger batching.
userType	userGroup	Type of the default assignee of this task. In this case it will be a usergroup.
userName	Standardusers	Name/Identifier of the Default assignee of the task. ==> The item will be assigned to the usergroup "Standardusers".
uiTemplate	Item approve UI	This is the default UI of the task.
executeDq	Never	The circumstances when a DQ should be run. In this case never!
nextStep	STEP:4	Identifier of the next step, The item will be moved to the next step, when this step is finished.

Step 4

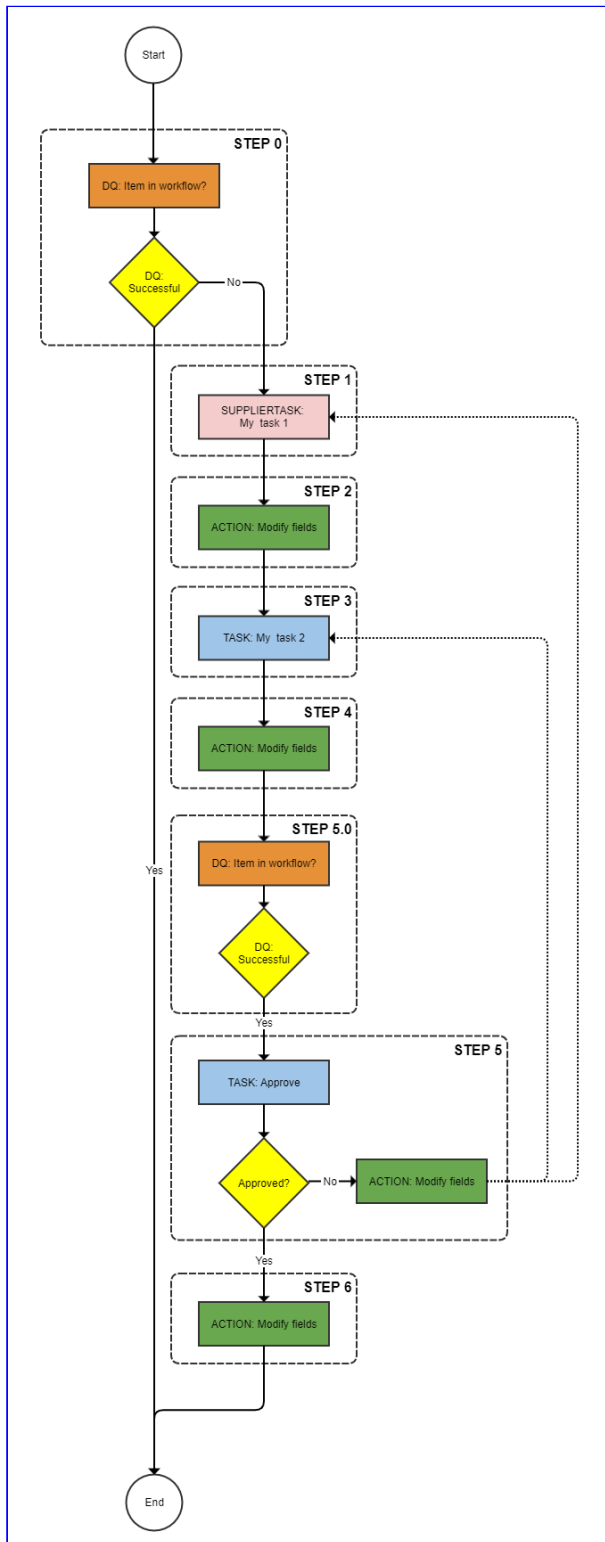
Step 4 is similar to step 2, but it will set another value to the field.

Key	Value	Description
id	4	Same keys and values than in the example above.
entity	Article	Entity container for this workflow.
updateFieldDescriptor	Article.CurrentStatus	This field allows the step to update a field in Product 360. The field must be fully qualified.
updateFieldValue	300	This is the value for the update.

3.3.14 Example 7 (2 Tasks + modifying of fields + approval step)

The next example extends the example 6 workflow with modifying of fields within the workflow after finishing the tasks followed by an approval task which also sets data if the task(s) are rejected.

3.3.14.1 Diagram for example 7



3.3.14.2 Product 360 prerequisites for example 7

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI

3.3.14.3 Explanation of the steps example 7

Step	Description
0	Check whether item is already in this workflow
1	Put the item into the supplier task "My task 1"
2	Modify the field Article.CurrentStatus to the value 200
3	Put the item into the task "My task 3"
4	Modify the field Article.CurrentStatus to the value 300
5.0	Check whether item is already in this workflow
5	Approve task with rejection options to all previous tasks. If rejected modify the field Article.CurrentStatus to the value 300
6	Modify the field Article.CurrentStatus to the value 400

3.3.14.4 StepWorkflow.xml file for example 7

Example 7

```
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 7</label>
    <identifier>Workflow_07</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
      <executeDq>Always</executeDq>
      <dqService>ItemsInWorkflowTasks-Process</dqService>
    </step>
  </workflow>
</imp:payload>
```

```

    <dqFailStep>STEP:1</dqFailStep>
    <getField>Article.MainSupplier->Party.Name</getField>
</step>
<step>
  <id>1</id>
  <entity>Article</entity>
  <workflowStatus>My task 1</workflowStatus>
  <description>My task 1</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>Always</enterStatus>
  <batchSize>500</batchSize>
  <userType>supplier</userType>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Never</executeDq>
  <getField>Article.CurrentStatus</getField>
  <nextStep>STEP:2</nextStep>
  <nextStep>STEP:3</nextStep>
</step>
<step>
  <id>2</id>
  <entity>Article</entity>
  <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
  <updateFieldValue>200</updateFieldValue>
</step>
<step>
  <id>3</id>
  <entity>Article</entity>
  <workflowStatus>My task 2</workflowStatus>
  <description>My task 2</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>Always</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Never</executeDq>
  <nextStep>STEP:4</nextStep>
  <nextStep>STEP:5.0</nextStep>
</step>
<step>
  <id>4</id>
  <entity>Article</entity>
  <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
  <updateFieldValue>300</updateFieldValue>
</step>
<step>
  <id>5.0</id>
  <entity>Article</entity>
  <enterStatus>Never</enterStatus>
  <batchSize>500</batchSize>
  <executeDq>Always</executeDq>
  <dqService>ItemsInWorkflowTasks-Process</dqService>

```

```

    <dqFailStep>STEP:5</dqFailStep>
  </step>
  <step>
    <id>5</id>
    <entity>Article</entity>
    <workflowStatus>Approve</workflowStatus>
    <description>Approval</description>
    <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
    <enterStatus>Always</enterStatus>
    <batchSize>500</batchSize>
    <userType>userGroup</userType>
    <userName>Standardusers</userName>
    <uiTemplate>Item approve UI</uiTemplate>
    <executeDq>Never</executeDq>
    <rejectTrigger>StepWorkflow_ParallelTasks-Reject</rejectTrigger>
    <singleChoice>true</singleChoice>
    <rejectDecision>
      <id>p360.bpm.reject.STEP:1</id>
      <label>Reject to task 1</label>
    </rejectDecision>
    <rejectDecision>
      <id>p360.bpm.reject.STEP:3</id>
      <label>Reject to task 2</label>
    </rejectDecision>
    <rejectFieldDescriptor>Article.CurrentStatus</rejectFieldDescriptor>
    <rejectFieldValue>100</rejectFieldValue>
    <nextStep>STEP:6</nextStep>
  </step>
  <step>
    <id>6</id>
    <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
    <entity>Article</entity>
    <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
    <updateFieldValue>400</updateFieldValue>
  </step>
</workflow>
</imp:payload>

```

3.3.14.5 Detailed explanation of the steps

Step 0 - Step 3

These steps are exact copy of the example 6.

Step 4

Step 4 is nearly the same as the step 4 of the example above. Only the differences are shown in the next table.

Key	Value	Description
...	...	Same keys and values than in the example 6.
nextStep	STEP:5.0	Identifier of the next step, The item will be moved to the next step, when this step is finished.

Step 5.0

This steps checks only whether the item is already in this workflow. If not the executed next step is 5. If yes the workflow will end to avoid that the item will go to the step 5 (Approve) until it is somewhere else in the workflow.

Key	Value	Description
id	5.0	Identifier of the step.
entity	Article	Entity container for this workflow.
enterStatus	Never	The circumstances when an item should enter a status (task). In this case never!
batchSize	500	Batch size to use with trigger batching.
dqService	ItemsInWorkflowTasks-Process	This DQ service is an additional bpel inside the StepWorkflow which checks whether the item is already inside this workflow or not.
dqFailStep	STEP:5	The item will be moved to the step with the identifier 5 if the above DQ fails.

Step 5

This is the approval step with modifying a field while rejecting.

Key	Value	Description
id	5	Identifier of the step.
entity	Article	Entity container for this workflow.
workflowStatus	Approve	Status name within the Product 360 workflow. ==> Product 360 task name = [Workflow] - [Status] - [Catalog]
description	Approval	Description of the status.
workflowServiceEndpoint	StepWorkflow-Trigger	This is the service name of the partner link. This name is defined in the process deployment descriptor of the workflow.

Key	Value	Description
enterStatus	Always	The circumstances when an item should enter a status (task). In this case always!
batchSize	500	Batch size to use with trigger batching.
userType	userGroup	Type of the default assignee of this task. In this case it will be a usergroup.
userName	Standardusers	Name/Identifier of the Default assignee of the task. ==> The item will be assigned to the usergroup "Standardusers".
uiTemplate	Item approve UI	This is the default UI of the task.
executeDq	Never	The circumstances when a DQ should be run. In this case never!
rejectTrigger	StepWorkflow_ParallelTasks-Reject	Endpoint for the reject trigger of parallel states (tasks).
singleChoice	true	This is for Rejects only and determines if you can have more than one choice. ==> The workflow can be rejected to only one step,
<rejectDecision>id	p360.bpm.reject.STEP:1	Identifier of the 1 st reject step.
<rejectDecision>label	Reject to task 1	Label of the 1 st rejection. This label will show up in the UI.
<rejectDecision>id	p360.bpm.reject.STEP:3	Identifier of the 2 nd reject step.
<rejectDecision>label	Reject to task 2	Label of the 2 nd rejection. This label will show up in the UI.
rejectFieldDescriptor	Article.CurrentStatus	This is the field the will be modified if the task will be rected. The field must be fully qualified.
rejectFieldValue	100	This is the value for the update.
nextStep	STEP:6	Identifier of the next step, The item will be moved to the next step, when this step is approved.

Step 6

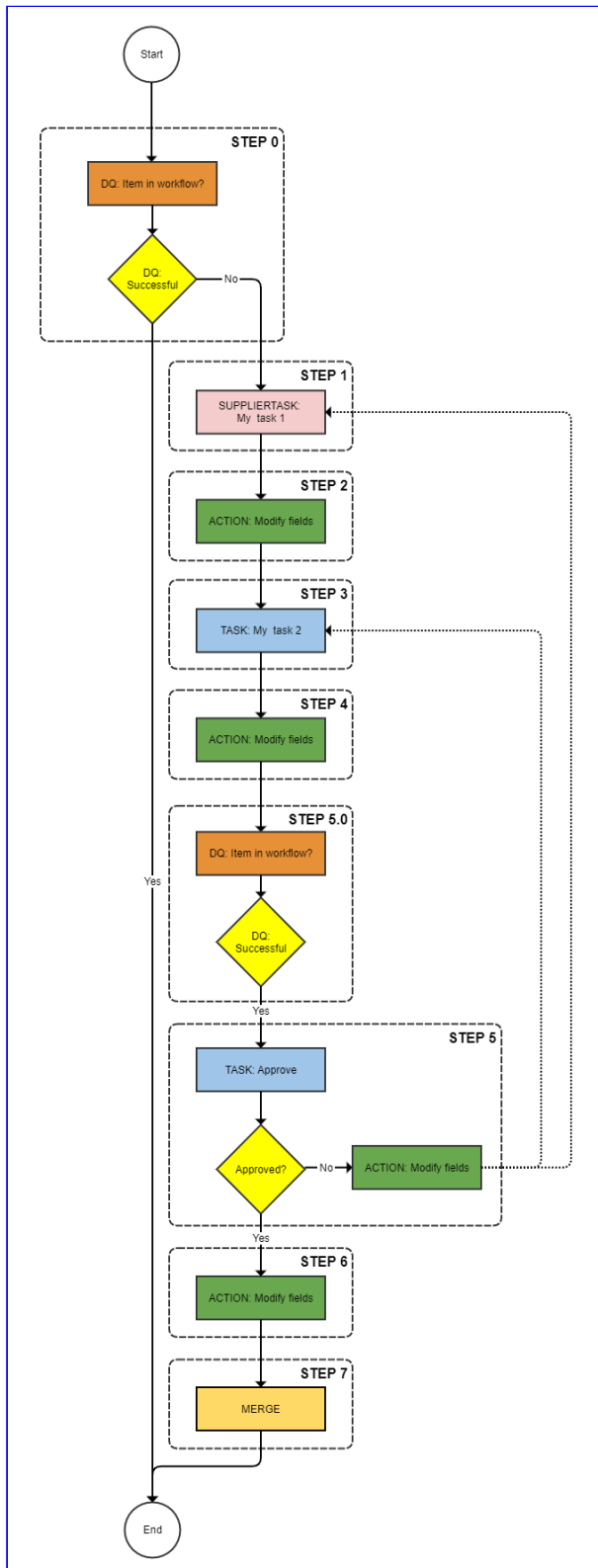
Step 2 will manipulate data within the workflow. In this example we will change the field CurrentStatus of the entity Article.

Key	Value	Description
id	6	Same keys and values than in the example 6.
entity	Article	Entity container for this workflow.
updateFieldDescriptor	Article.CurrentStatus	This field allows the step to update a field in Product 360. The field must be fully qualified.
updateFieldValue	400	This is the value for the update.

3.3.15 Example 8 (2 Tasks modifying of fields approval step merge)

The next example extends the example 7 workflow with merging the item(s) after the approval.

3.3.15.1 Diagram for example 8



3.3.15.2 Product 360 prerequisites for example 8

Product 360 entity	value
Usergroup	Standardusers
Ui Template	Item approve UI

3.3.15.3 Explanation of the steps example 8

Step	Description
0	Check whether item is already in this workflow
1	Put the item into the supplier task "My task 1"
2	Modify the field Article.CurrentStatus to the value 200
3	Put the item into the task "My task 3"
4	Modify the field Article.CurrentStatus to the value 300
5.0	Check whether item is already in this workflow
5	Approve task with rejection options to all previous tasks. If rejected modify the field Article.CurrentStatus to the value 300
6	Modify the field Article.CurrentStatus to the value 400
7	Merge the item

3.3.15.4 StepWorkflow.xml file for example 8

Example 8

```
<imp:payload xmlns:imp="http://www.informatica.com/schema/ItemMap" contentType="string">
  <workflow>
    <label>Example Workflow 8</label>
    <identifier>Workflow_08</identifier>
    <version>1.0</version>
    <step>
      <id>0</id>
      <entity>Article</entity>
      <enterStatus>Never</enterStatus>
      <batchSize>500</batchSize>
    </step>
  </workflow>
</imp:payload>
```

```

    <executeDq>Always</executeDq>
    <dqService>ItemsInWorkflowTasks-Process</dqService>
    <dqFailStep>STEP:1</dqFailStep>
    <getField>Article.MainSupplier->Party.Name</getField>
</step>
<step>
  <id>1</id>
  <entity>Article</entity>
  <workflowStatus>My task 1</workflowStatus>
  <description>My task 1</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>Always</enterStatus>
  <batchSize>500</batchSize>
  <userType>supplier</userType>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Never</executeDq>
  <getField>Article.CurrentStatus</getField>
  <nextStep>STEP:2</nextStep>
  <nextStep>STEP:3</nextStep>
</step>
<step>
  <id>2</id>
  <entity>Article</entity>
  <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
  <updateFieldValue>200</updateFieldValue>
</step>
<step>
  <id>3</id>
  <entity>Article</entity>
  <workflowStatus>My task 2</workflowStatus>
  <description>My task 2</description>
  <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
  <enterStatus>Always</enterStatus>
  <batchSize>500</batchSize>
  <userType>userGroup</userType>
  <userName>Standardusers</userName>
  <uiTemplate>Item approve UI</uiTemplate>
  <executeDq>Never</executeDq>
  <nextStep>STEP:4</nextStep>
  <nextStep>STEP:5.0</nextStep>
</step>
<step>
  <id>4</id>
  <entity>Article</entity>
  <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
  <updateFieldValue>300</updateFieldValue>
</step>
<step>
  <id>5.0</id>
  <entity>Article</entity>
  <enterStatus>Never</enterStatus>
  <batchSize>500</batchSize>

```

```

    <executeDq>Always</executeDq>
    <dqService>ItemsInWorkflowTasks-Process</dqService>
    <dqFailStep>STEP:5</dqFailStep>
  </step>
  <step>
    <id>5</id>
    <entity>Article</entity>
    <workflowStatus>Approve</workflowStatus>
    <description>Approval</description>
    <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
    <enterStatus>Always</enterStatus>
    <batchSize>500</batchSize>
    <userType>userGroup</userType>
    <userName>Standardusers</userName>
    <uiTemplate>Item approve UI</uiTemplate>
    <executeDq>Never</executeDq>
    <rejectTrigger>StepWorkflow_ParallelTasks-Reject</rejectTrigger>
    <singleChoice>true</singleChoice>
    <rejectDecision>
      <id>p360.bpm.reject.STEP:1</id>
      <label>Reject to task 1</label>
    </rejectDecision>
    <rejectDecision>
      <id>p360.bpm.reject.STEP:3</id>
      <label>Reject to task 2</label>
    </rejectDecision>
    <rejectFieldDescriptor>Article.CurrentStatus</rejectFieldDescriptor>
    <rejectFieldValue>100</rejectFieldValue>
    <nextStep>STEP:6</nextStep>
  </step>
  <step>
    <id>6</id>
    <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
    <entity>Article</entity>
    <updateFieldDescriptor>Article.CurrentStatus</updateFieldDescriptor>
    <updateFieldValue>400</updateFieldValue>
    <nextStep>STEP:7</nextStep>
  </step>
  <step>
    <id>7</id>
    <workflowServiceEndpoint>StepWorkflow-Trigger</workflowServiceEndpoint>
    <entity>Article</entity>
    <mergeProfile>profile_no_images</mergeProfile>
    <nextStep>MergeItems-Process</nextStep>
  </step>
</workflow>
</imp:payload>

```

3.3.15.5 Detailed explanation of the steps

Step 0 - Step 5

These steps are exact copy of the example 7.

Step 6

Step 6 is nearly the same as the step 5 of the example above. Only the differences are shown in the next table.

Key	Value	Description
...	...	Same keys and values than in the example above.
nextStep	STEP:7	Identifier of the next step, The item will be moved to the next step, when this step is finished.

Step 7

Step 7 will execute the merge.

Key	Value	Description
id	7	
workflowServiceEndpoint	StepWorkflow-Trigger	
entity	Article	
mergeProfile	profile_no_images	Profile of the merge process. There are 2 profiles within the standard workflow package. They are located in the folder profiles merge-profile of the StepWorkflow project. If you need additional profiles you have to add them to the project with your BPM Designer ,
nextStep	MergeItems-Process	This bpel process allows you to merge items into the Master Catalog.

3.4 Creating MQ based workflows

This guide describes how to design ActiveVOS workflows that leverage the new queue based communication mode between P360 and ActiveVOS. Consult the Installation and Operation guide for the following areas:

- Configuration of queues in P360 server
- Installation of ActiveVOS with ActiveMQ support and configuration of queues in ActiveVOS (*Note: ActiveVOS version 9.2.4.6 is a prerequisite for queue based interactions between P360 and ActiveVOS*)
- Message headers and message formats for the various interactions between P360 and ActiveVOS

Table of Contents

- [REST versus JMS based workflows](#)(see page 505)
- [Mandatory Permission Required For Queue Based Communication](#)(see page 506)
- [Types of interactions and their representation in the WSDL file](#)(see page 506)
- [Router process](#)(see page 508)
- [Core orchestration project](#)(see page 508)
- [Correlation](#)(see page 509)
- [Defining participants](#)(see page 509)
 - [Consumers](#)(see page 509)
 - [Producers](#)(see page 509)
- [Workflow interaction examples](#)(see page 511)
 - [Workflow instance creation](#)(see page 511)
 - [P360 Service API oneway calls](#)(see page 512)
 - [P360 Service API calls with result handling](#)(see page 512)
 - [P360 DQ execution with result handling](#)(see page 514)
 - [P360 workflow tasks](#)(see page 514)

3.4.1 REST versus JMS based workflows

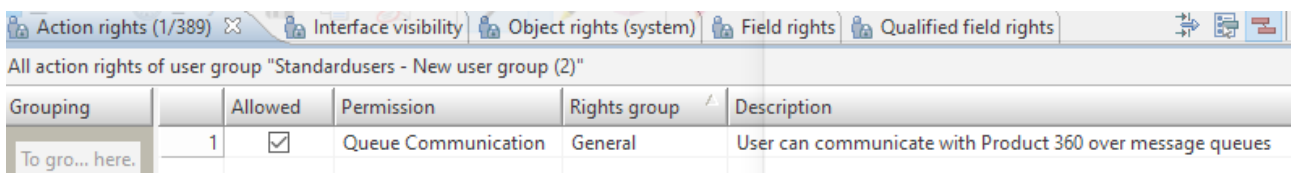
At its core, ActiveVOS is based on SOAP Style web services backed by WSDL definitions. The support for both REST based web services and for JMS interactions are internally mapped onto these concepts. The details are slightly different and they have an impact on how interactions with P360 need to be added to workflows.

Area	REST based communication	Queue based communication
Service interface definitions	Leverages built-in generic REST service interface definitions.	Interface definitions need to be added as a WSDL in the ActiveVOS project.
Message header	Full access to REST headers through the built in interface definition.	Access to JMS header fields is available through "Partner" variables.
Message body	Free choice of format, can be JSON.	Message body needs to adhere to WSDL and therefore restricted to XML.
Incoming messages	Usually paired with an outgoing message to implement the HTTP response.	Oneway, there is no need to send a response message to the queue. P360 regards the message as successfully delivered as soon as it has been placed into the queue.

Area	REST based communication	Queue based communication
Outgoing messages	Outgoing messages may be "one-way" or paired with incoming messages that contain the response for the outgoing request,	Outgoing messages are written to a queue. Similarly to the REST case this may be paired with an incoming message for the response which also comes in via a queue. However from a workflow designer perspective they can be seen as two "one-way" communications.
Special case of outgoing message: Workflow tasks definition	Body can be either JSON or XML. Workflow task definitions include ActiveVOS workflow endpoints that have to be invoked via REST.	Body has to be XML. Workflow task definitions set the communication mode to "QUEUE", include the queue ID (as defined in P360 configuration) and the workflow endpoints to be invoked via JMS.

3.4.2 Mandatory Permission Required For Queue Based Communication

For BPM to have queue based communication with P360, "Queue Communication" named Action rights is required for the user group of the user used in BPM workflows. Without having this action right for the user group, BPM will get Unauthenticated error when it try to contact Product 360 via message queue.



All action rights of user group "Standardusers - New user group (2)"					
Grouping		Allowed	Permission	Rights group	Description
To gro... here.	1	<input checked="" type="checkbox"/>	Queue Communication	General	User can communicate with Product 360 over message queues

3.4.3 Types of interactions and their representation in the WSDL file

There are a couple of different message types that can be received from/sent to P360:

- Incoming event triggers
- Outgoing Service API requests (one-way or request-response)
- Incoming responses for Service API requests (result or error responses)
- Outgoing DQ execution requests
- Incoming responses for DQ executions (result or error responses)

To avoid the overhead of defining each of these messages down to the smallest detail within the WSDL file (or a separate XML Schema file) all of the messages have been defined to have [anyType](#) content within some root tag. The approach is different between incoming and outgoing messages:

- For *incoming* messages we can use a *single type* with an arbitrarily defined root tag as ActiveVOS is not strictly checking the root tag.
- For *outgoing* messages the defined root tag is what gets sent to P360 and needs to match P360 expectations. Therefore there are *a number of message types defined with different root tags*.

Based on these messages several port types have been defined:

Port type	Operation	Input message type	Remarks
Receive	receive	p360RoutedMessage	<p>This message type has two parts:</p> <ul style="list-style-type: none"> • p360Header: this part for now only contains an optional element <code>correlationId</code> • p360MessageBody: this part contains the original message sent from P360 (with an adjusted root tag)
Send	callWithoutParameters	emptyMessage	Used for sending requests to P360 that do not require any input parameters. Due to WSDL restrictions the operation still needs to specify a message type.
	startWorkflow	startWorkflowMessage	This operation and the following ones are used for requests that require different types of input messages.
	enterWorkflowStatus	enterWorkflowStatusMessage	
	leaveWorkflowStatus	leaveWorkflowStatusMessage	
	executeQualityProfile	dataQualityProfileMessage	
	listWrite	entityItemTableMessage	
Router	route	p360Message	Only used by the P360Router process - see next section.

3.4.4

Router process

In the REST based communication mode P360 invokes the exposed REST endpoint of the respective ActiveVOS process directly. In the queue based communication mode all P360 requests go to a single queue.

To deal with this a router process is set up on ActiveVOS which has essentially three responsibilities;

- Evaluate the JMS header property `P360TargetService` that is used to forward the request to the right endpoint.
- Copying the request message into a common base format (`p360RoutedMessage`) that all endpoints can deal with.
- Adding the correlation ID to the common base format message (see below for details).



When ActiveVOS monitors a queue for incoming messages it supports reading the target endpoint from the JMS header property `JmsTargetService`. However this feature does not work with endpoints that are based on `anyType` messages.

3.4.5 Core orchestration project

As described above, both the WSDL and the P360Router process are key components for setting up queue based communication between P360 and ActiveVOS. Therefore a ready-to-use orchestration project is made available as a ZIP file. It contains the WSDL and the BPEL and PDD files for the P360Router. Import this project into your ActiveVOS Designer workspace. While you can then add your workflow process definitions simply to this project, it is recommended to set them up separately:

- Create one or more separate projects for your workflow processes.
- Add the P360_JMS_Core project as a project reference in your project(s).
- Use the message, port and partner link types of the WSDL when defining variables and participants in your processes.
- The WSDL also includes a correlation definition that can be used in the correlation set definitions of your workflow processes (see below).

The P360_JMS_Core project needs to be deployed to your ActiveVOS as a prerequisite for the deployment of your own workflows. Before doing so note the endpoint name of the "Consumer" participant of the P360Router process in the PDD file. This name needs to match the default service set of the queue listener set up in the ActiveVOS Message Service configuration. By default both are called *P360RouterService*.

The listener monitors the queue which P360 uses to send requests to ActiveVOS. These messages need to flow through the P360Router process before they are forwarded to the actual target endpoint.

The core orchestration project is included as a ZIP file in the BPM accelerator package.

3.4.6 Correlation

The correlation concept assures that incoming messages are routed to the right process instance of a workflow. A good starting point is to use the process ID for this, however for more complex workflows with parallel execution it may be necessary to add additional information to the correlation IDs of specific endpoints within the workflow.

i ActiveVOS sets the correlation ID by picking it from a defined path of an outgoing message. It then checks a defined path in incoming messages to map the request to an existing process instance. For this to work the correlation ID must be part of both message bodies.

For backwards compatibility reasons many P360 calls expect the correlation ID to be sent in the `processId` field of a request and also return it in the same way in subsequent requests to ActiveVOS. Newer APIs however support the use of the `JMSCorrelationID` header.

On the receiving side the `P360Router` process takes care of picking the correlation ID from the right location and puts it into the `p360Header` part of the `p360RoutedMessage`. So the individual workflows can define their correlation ID settings based on this without worrying how it was transmitted.

Unfortunately on the outgoing side these details cannot be hidden. Best practice here is to:

- Write the correlation ID to the `JMSCorrelationID` header of the outgoing message. This is where P360 expects it in general.
- Write it to a top level element called `processId` in the outgoing message body. This is for ActiveVOS correlation definition to be able to pick it up on the way out.
Note that the name `processId` is used for backwards consistency. This is where the correlation ID is already passed in the REST based communication use case.

3.4.7 Defining participants

3.4.7.1 Consumers

Configure the Consumer participants in your workflow processes as follows

- Select `p360ReceivePortType` as its service interface.
- Select `p360ReceivePLT` as its partner link type
- Add a receive activity to your workflow, select the newly created Consumer, select the `receive` operation and use a variable of type `p360RoutedMessage` to store the input.
- After creating the deployment descriptor for your workflow set "Binding" to "Document literal" and the name of the endpoint to how it should be addressable from P360.

Note that such a Consumer is always expecting incoming messages at its defined endpoint. So usually you *cannot reuse* the same Consumer at different points in your workflow.

3.4.7.2 Producers

Configure a Producer participant as follows:

- Select p360SendPortType as its service interface.
- Select p360SendPLT as its partner link type
- After creating the deployment descriptor for your workflow set the Producer to be a "JMS Service" and "dynamic"
- Prepare the an input variable and the Producer in a script activity of your workflow
 - Setup a variable of the right type for the desired operation's input message body.
 - Setup the message header by assigning an appropriate XML structure to your newly created "Partner" (Choose "Partner" and your new participant as the target of a copy).
See section below on how to create this "appropriate XML".
 - Add an invoke activity to your workflow, select the newly created participant, the desired operation and the variable as its input.

Note that with the "dynamic" setting the same Provider can be used for any outgoing calls from the workflow to P360. You just need to prepare the header and the input variable upfront.
In case of parallel execution threads you should still use separate Provider participants.

Setting up the outgoing message header

- Add a script activity to your workflow
- Add a copy assignment to the script activity
 - On the source side select "Expression" and set it to XQuery based expressions.
 - On the target side select "Partner" and the newly created Producer
- Go back to the PDD file and temporarily select "static" for the Producer. This will populate the text box below with an XML template. Copy this for further use.
- Go back to the copy assignment and paste the copied XML as the starting point for the expression. You need to make the following adjustments:

Section	Value
address	The address needs to be set to the name of the target queue. Note that it needs to be the JNDI name configured for the queue on the ActiveVOS server side - not the actual queue name on the JMS server. There are two distinct queues that can be used here at the moment: one for DQ executions and one for Service API requests.
jmsManagerId	The jmsManagerId needs to match the name of the "Message Service" configuration in the ActiveVOS server.
jmsMessageFormat jmsMessageType	Leave these at "xml", "text" respectively.
jmsTTL	Set to "0" for most cases to avoid message expiration. Sometimes messages can become obsolete if they haven't been dealt with in a defined time frame. In such cases you can set this to the maximum number of milliseconds that this message should be kept on the queue. See the documentation of your JMS server on how expired messages are treated.

Section	Value
ReferenceParameters	<p>This is essentially a list of key value pairs that will appear as header properties on the JMS message. Common properties for any type of message include:</p> <ul style="list-style-type: none"> • User - P360 user name • Password - password for the P360 user • JMSCorrelationID for two way communication. P360 will include this as a header property in the response. • SuccessTargetService - for two way communication this specifies the value of the P360TargetService header property in the response. • ErrorTargetService - for two way communication this can be used optionally to specify a different for P360TargetService in error responses. <p>Individual message types may require more properties.</p>



The WSDL includes partner link types. When adding new participants to a BPEL process and selecting a port type for its service interface ActiveVOS automatically creates a separate partner link type in a new WSDL. You can see these new WSDLs pop up under wsdl/bpel in your project structure. While it is not strictly necessary you can change the partner link types of your participants to use the types from the Core WSDL instead. Once you have done this the generated WSDLs should be empty as ActiveVOS Designer realizes you don't need the generated partner link types any more. However they are not removed completely and still appear as imports in your projects. To completely get rid of them, remove the imports first and then the WSDL files. Always make sure the files just contain an empty root tag before doing this. Also, as the behavior is the same for each new participant you might want to only do this once you have all participants defined.

3.4.8 Workflow interaction examples

Workflow examples are provided in two separate orchestration projects - P360_JMS_Demo and P360_JMS_Demo_DataQuality. The later one contains a DQ execution specific example and the other one has all the other examples. Separation into two projects was arbitrary and just to show that workflow processes can be defined in several projects. Both projects rely on the P360_JMS_Core project for reference to the WSDL and also at run time for the routing functionality.

3.4.8.1 Workflow instance creation

The **NoOpReceiver** example workflow contains a minimal workflow. It basically contains a consumer, a variable and a receive activity that assigns the incoming message to the variable. Instances of it will appear in the ActiveVOS monitoring for each message that was successfully received.

Use this workflow for testing your setup for incoming messages:

- To test ActiveVOS Message Service setup and P360Router process deployment publish a message to the queue that is monitored by ActiveVOS
 - Message Body can be arbitrary XML.
 - Message header should include P360TargetService property set to NoOp-Trigger .
 - Instances of the workflow should appear in monitoring and the process variable should contain the message body sent.
- To test P360 to ActiveVOS communication set up a change trigger in P360 that is sending messages via queue to NoOp-Trigger.
 - Instances of the workflow should appear in monitoring and the process variable should contain the message sent from P360.

3.4.8.2 P360 Service API oneway calls

Note: This example is using the LIST API to issue a write request to P360. The queue based communication support for the write direction of the LIST API is experimental in the current release.

The **ServiceApiCallOneway** example workflow shows how to invoke P360 from workflows when there is no need to receive a response from P360. Here is a short summary of its content:

- It consumes incoming trigger messages at the endpoint ServiceApiCallOneway-Trigger .
 - Again, for testing this workflow it should be hooked up with a change trigger definition on P360 side. E.g. whenever the short description of an item changes.
- Additionally it contains a Provider and an invoke activity using it for sending a message to the P360 Service API.
 - The concrete call is setting the CurrentStatus of the item extracted from the received trigger message to "07 Attributes OK"

Compared to the generic header property description above this call needs additional properties:

Property	Value	Description
P360Url	rest/V2.0/list/Article	Service API endpoint to invoke on P360 side.
Method	POST	Does not strictly have to be set as this is the default. Don't be confused if this looks like REST/HTTP. The communication is still JMS based but it is mapped onto the existing REST based API on P360 side.

You can use this workflow to test the full roundtrip P360->ActiveVOS → P360. The trigger message is expected to be an entity change trigger containing the entity type and id. The workflow extracts this information from the incoming message and uses it within the message body of the outgoing request. Set up a change trigger on P360 that reacts on changes to the "short description" on items. Then create some items - they will have their initial "status" set to "01". Changing the short description should lead to the creation of a workflow instance and then to call to P360 that changes the status of the item to "07".

3.4.8.3 P360 Service API calls with result handling

The **ServiceApiCall** example workflow shows how to invoke P360 Service API and process the response.

Compared to the one-way example we need two additional Consumer participants, one for the success case and one for the error case. The example uses the "Pick" activity to wait on two endpoints at the same time. Additionally it also has a "timeout" branch defined. There are three possible outcomes:

- Response is received on the endpoint waiting for "success" responses → workflow will be shown as completed in monitoring.
- Response is received on the endpoint waiting for "error" responses → workflow will be shown as faulted in monitoring.
- No response is received within the time set up on the "timeout" branch → workflow will be shown as faulted in monitoring.

This is a simple example. In the real world the workflow might continue in each of these cases but take different decisions based on the outcome of this call.

Note that for the correlating requests a correlation set has been defined using the definition provided in the WSDL. This correlation set is used on the outgoing "invoke" activity and the matching "receive" activities.

The outgoing message has the following header properties set:

Property	Value	Description
P360Url	rest/V2.0/list/Article/byCatalog	Service API endpoint for listing the items in the master catalog.
Method	GET	We want to the functionality associated with the GET method in this endpoint.
ResponseQueueID	bpm	This is the ID of the message queue to which the response should be written as configured in P360 service properties (not the JNDI name used in ActiveVOS!)
SuccessTargetService	ServiceApiCall-Response	The endpoint to invoke in case of successful execution. The response message will use this as the value for the P360TargetService property.
ErrorTargetService	ServiceApiCall-Error	The endpoint to invoke in case of a failure during execution. The response message will use this as the value for the P360TargetService property.
JMSCorrelationID	...	Correlation id to match up response with the right workflow instance in ActiveVOS.

Special handling of GET requests

REST/HTTP GET calls do not require/support an input message body. However the request and response are mapped here onto two distinct operations defined in the WSDL and each operation needs an input message:

- The request is using the `callWithoutParameters` and an `emptyMessage` as input.
- The response is using the standard `receive` operation and the `p360RoutedMessage`.

So the `emptyMessage` is essentially a workaround for having to provide a message type for every operation. But there is also a different reason for having it. As described in the section on correlation handling the ActiveVOS engine needs the correlation ID to be part of the outgoing message. So in cases like this we still have to add it to the `emptyMessage`.

3.4.8.4 P360 DQ execution with result handling

The **CallDataQuality** workflow example shows how to execute DQ rule configurations. From a structural perspective it is very similar to the previous example that calls the Service API and waits for the response:

- A total of three consumers: initial trigger, success response, error response
- A provider for invoking P360 endpoint
- A "Pick" activity for reacting on the response from P360
- Correlation ID present in both the `JMSCorrelationID` header property and outgoing message body. Correlation set defined and used on "invoke" and matching "receives".

There are however also some differences:

- P360 listens for these requests on a separate queue, so the "address" of the Provider participant needs to be set to the correct JNDI name (`JNDI_P360_DATA_QUALITY`).
- The Service API specific parameters are not needed: `P360Url`, `Method`

Note that this workflow assumes that there is a DQ rule configuration present on P360 which is named `IsEmpty`. It should be defined so that it checks whether the short description of an item is filled or not. For testing add a change trigger on short description changes and invoke `CallDataQuality-Trigger` endpoint. Add the short description and the DQ status of this rule configuration to the table in the P360 rich client and observe how filling/emptying the short description of an item changes the DQ status.

3.4.8.5 P360 workflow tasks

The **WorkflowTasks** workflow example shows how to set up workflow tasks on P360. P360 workflow tasks can be used to include manual tasks by users into the overall workflow.

Workflow task management calls are part of the Service API, so the explanations in the previous examples around Service API calls apply here as well.

The overall structure of the workflow is as follows:

1. Receive a trigger to start the workflow - expected to be an entity create/change trigger.
2. Invoke Service API to define the workflow task structure - our case the definition includes two steps/statuses: `status01` and `status02`
 - a. Note that this is done for each instance of the workflow process. If P360 already knows about the workflow task definition it will ignore this. However the workflow could be sending a newer version of the definition which will cause an update on the P360 side.
3. Tell P360 that the item extracted from the trigger message should "enter" `status01`.
4. Wait for message from P360 that states that some user has marked `status01` as finished for this item.
5. Tell P360 that the item extracted from the trigger message should "leave" `status01`.
6. Tell P360 that the item extracted from the trigger message should "enter" `status02`.
7. Wait for message from P360 that states that some user has marked `status01` as finished for this item.

8. Tell P360 that the item extracted from the trigger message should "leave" *status02*.

Note that steps 4 and 7 are waiting for messages at certain endpoints: *WorkflowTasks-Finish01* and *WorkflowTasks-Finish02*. P360 knows to use these endpoints in the *P360TargetService* header because they are included in the workflow task definition sent to P360 in step 2.

Correlation is needed between steps 4 and 5 and again between 7 and 8. For this purpose a correlation set is set up similarly to previous examples.

4 User Interface Templates

4.1 Content

This chapter describes how you can import the different standard User Interface Templates (UI Templates). The examples are located in the accelerator package in the sub folder "UI_Templates".

4.2 Import

- Start the application "PIM Desktop client".
- Select the menu entry "Manage UI templates" in the menu "Management".
- Click on the button "Load UI templates from files...".
- Select a xml file and click ok.

After a successful import you can make also some changes for your system, for example which classification system will be used in the classification UI. For further information please read the configuration manual.

4.3 Approval UI

The Approval UI allows to quickly check the details of a list of objects in order to approve them in context of a specific work step. The object detail view on the right can be tailored to showcase the specific information needed for the approval of a given work step.

Filename	Location	Entity
Item approve UI.xml\Examples\Flexible UI	item
Product approve UI.xml\Examples\Flexible UI	product
Variant approve UI.xml\Examples\Flexible UI	variant

Task: Approve item commercial data (59 Item(s) - Accepted)

Image	Item no.	Status	Short description
	Article_2918766499071457	01 New	Remoty Remote Control
	Article_2918766499071461	01 New	DIGIT ISIO S1, black
	Article_2918766499071464	01 New	TechniControl, black
	Article_2918766499071467	01 New	DigiCorder HD K2, 160 GB, black
	Article_2918766499071470	01 New	TechniStar S2+, black with HD+ Smartcard
	Article_2918766499071473	01 New	TechniBox HD VA
	Article_2918766499071476	01 New	Comfort remote control (HD-Vision), black
	Article_2918766499071481	01 New	DIGIT ISIO S1, silver
	Article_2918766499071484	01 New	TechniControl Plus, silver
	Article_2918766499071487	01 New	HOT 4, bicoloured
	Article_2918766499071490	03 Selling pr	TechniStar K1, black
	Article_2918766499071493	01 New	DIGIT ISIO S, black
	Article_2918766499071496	01 New	TechniBox K1, black
	Article_2918766499071499	01 New	DIGYBOX HD C+, black
	Article_2918766499071502	01 New	DIGIT ISIO S, silver
	Article_2918766499071505	01 New	TechniControl Plus, silver (for Receiver)
	Article_2918766499071508	01 New	DigiCorder HD K2, 160 GB, silver
	Article_2918766499071511	01 New	DIGIT ISIO C, black
	Article_2918766499071514	01 New	TechniBox HD VAC, silver
	Article_2918766499071517	01 New	DIGYBOX HD CX, black
	Article_2918766499071520	01 New	Connection cable (HDMI)
	Article_2918766499071525	01 New	Remoty Plus, silver
	Article_2918766499071528	01 New	DigiCorder HD S3
	Article_2918766499071531	01 New	TechniBox K1, black

Detail (Item)

Short description:
TechniStar K1, black

Long description:
Digital HDTV cable receiver (MPEG-2/MPEG-4) with integrated CONAX conditional access system, smartcard reader and CI+

Item no.:
Article_2918766499071490

GTIN:
00083250028171

Manufacturer:
TechniSat

Net customer price(\$):
199.99

Non-binding price recommendation(\$):
249.95

Gross list price(\$):
159.99

Net list price(\$):
199.96

Set Status:
03 Selling prices OK
01 New
02 Purchase prices OK
03 Selling prices OK
04 Commercial data OK
05 Initial classification OK
06 Long description OK
07 Attributes OK
08 Internet image OK
09 Print image OK
1-9/10

4.4 Text Mastering UI

Filename	Location	Entity
Item translation UI.xml\Examples\Flexible UI	item
Product translation UI.xml\Examples\Flexible UI	product
Variant translation UI.xml\Examples\Flexible UI	variant
Product with Items translation.xml\Examples\Flexible UI	product + item

The text mastering UI allows for objects assigned to a task to select them and easily get an overview of all marketing text relevant information assigned. Furthermore it is possible to compare and edit text values for different languages or target markets directly within the details section.

informatica

Task: Translate marketing texts for German target market (59 Item(s) - Accepted)

Name: Translate marketing texts for German target market

Assigned user: Andreeva, Tatyana

Escalation on: No content

Description: All items from the supplier catalog "Master catalog"

User group: No content

Time expires on: No content

Created by: Andreeva, Tatyana

Delegate: No content

Anticipated completion on: No content

Created on: 8/4/2015 5:07 PM

Responsible: Andreeva, Tatyana

Template: Item translation UI

Items (59)

Image	Item no.	Status	Short description
	Article_2918766499071452	01 New	Comfort remote control (HDTV Series)
	Article_2918766499071457	01 New	Remoty Remote Control
	Article_2918766499071461	01 New	DIGIT

Detail (Item)

Short description:
DIGIT ISIO S1, black

Long description:
HDTV digital satellite receiver with twin tuner, digital video recorder and Internet functionality via USB to record TV and radio programs on an external hard drive

Item no.:
Article_2918766499071461

GTIN:
00083250028168

Manufacturer:
TechniSat

Net customer price(\$):
369.99

Non-binding price recommendation(\$):
499.95

Gross list price(\$):

Translate

Languages: English

Short description: DIGIT ISIO S1, black

Long description: HDTV digital satellite receiver with twin tuner, digital video recorder and Internet functionality via USB to record TV and radio programs on an external hard drive

Keywords: receiver, dvb-s, satellite, home entertainment

Other remarks: Texts for US target market

Languages: German

Short description: DIGIT

Long description: No content

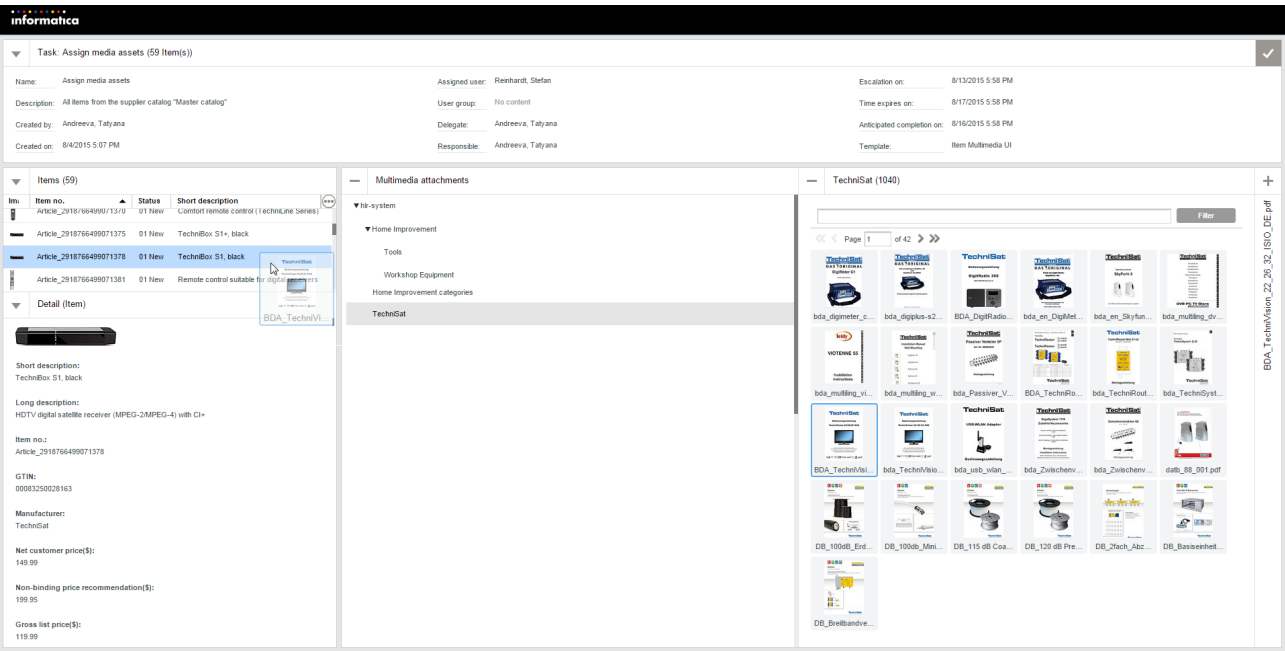
Keywords: No content

Other remarks: No content

4.5 Media Assignment UI

The media assignment UI makes assigning media assets to objects of a task more easy than ever before. The template combines a list of objects with the document categories tree and allows the user to drag and drop media assets directly on each of the objects from the list.

Filename	Location	Entity
Item multimedia assignment UI.xml\Examples\Flexible UI	item
Product multimedia assignment UI.xml\Examples\Flexible UI	product
Variant multimedia assignment UI.xml\Examples\Flexible UI	variant



4.6 Classification UI

The classification UI allows a very effective way of classifying product data. The user may select one or more objects assigned to the task to simply drag and drop them onto the structure group they should be assigned to.

Filename	Location	Entity
Item classification UI.xml\Examples\Flexible UI	item
Product classification UI.xml\Examples\Flexible UI	product
Variant classification UI.xml\Examples\Flexible UI	variant

Task: Classify items to structure system (59 Item(s) - Accepted)

Image	Item no.	Status	Short description
	Article_2018766499071317	01 New	Remote control suitable for digital receivers
	Article_2018766499071318	01 New	Remote control suitable for digital receivers
	Article_2018766499071326	01 New	Remoty Remote Control
	Article_2018766499071330	01 New	RS 232 Update Cable
	Article_2018766499071335	01 New	Remote control suitable for digital receivers
	Article_2018766499071340	01 New	External IR-receiver, for receiver with IR-port
	Article_2018766499071346	01 New	TechniStar S1, black
	Article_2018766499071349	01 New	DigiCorder HD S3
	Article_2018766499071352	01 New	ISIOControl Keyboard, black
	Article_2018766499071355	01 New	Remote control suitable for digital receivers
	Article_2018766499071358	01 New	USB Ethernet adapter
	Article_2018766499071364	01 New	SCART cable
	Article_2018766499071370	01 New	Comfort remote control (TechniLine Series)
	Article_2018766499071375	01 New	TechniBox S1+, black
	Article_2018766499071378	01 New	TechniBox S1, black
	Article_2018766499071381	01 New	Remote control suitable for digital receivers
	Article_2018766499071386	01 New	TechniControl, black
	Article_2018766499071389	01 New	LCD panel cleaner
	Article_2018766499071392	01 New	TechniSat TELTRONIC USB-WLAN
	Article_2018766499071395	01 New	TechniStar S2, black
	Article_2018766499071398	01 New	Remote control suitable for digital receivers
	Article_2018766499071403	01 New	HDT 4, black
	Article_2018766499071406	01 New	TechniStar IR, black

Structure

- enter
- ▼ ECLASS-5.0.1
- Construction technology
- Electric engineering, automation, process control engineering
- ▼ Information, communication and media technology
- ▼ Entertainment electronics
- Entertainment electronics (maintenance, inspection)
- ▼ Entertainment electronics (other)
- Entertainment electronics (other, unclassified)**
- Entertainment electronics furniture (other)
- Entertainment electronics (parts)
- Entertainment electronics (repair)
- Entertainment electronics furniture
- Machine, apparatus
- Machine, device (for special applications)
- Marketing
- Office products, facilities and technics, papelerie
- Service
- Tool

Image	Item no.	Short description
	Article_2018766499071317	Remote control suitable for digital receivers
	Article_2018766499071318	Remote control suitable for digital receivers
	Article_2018766499071326	Remoty Remote Control

Classified in Structure (3)

Article_2018766499071317 01 New Remote control suitable for digital receivers

Article_2018766499071318 01 New Remote control suitable for digital receivers

Article_2018766499071326 01 New Remoty Remote Control

Detail (Item)

4.7 Attribute Mastering UI

Filename	Location	Entity
Item attributes UI.xml	...\Examples\Flexible UI	item
Product attributes UI.xml	...\Examples\Flexible UI	product
Variant attributes UI.xml	...\Examples\Flexible UI	variant

The attribute mastering UI focuses on attribute values of objects attached to a task. By selecting an object the detail view gives an overview of all its attributes and their values so that a user can easily approve or update them.

5 Web Search Examples

There are six templates provided with the package as examples. These templates can be loaded in the export module of Product 360 Desktop client and used for exporting data to Elasticsearch for indexing. These templates provide examples of each type of field that can be indexed in Elasticsearch.

5.1 Items-only.ext

This template can be used when user wants to create an index for items from any one catalog. Catalog can be selected while exporting.

5.2 Products-only.ext

This template can be used when user wants to create an index for products only of master catalog in 1PPD , 2PPD or 3PPD system.

5.3 Variant.only.ext

This template can be used when user wants to create an index for variant only of master catalog in 1PPD , 2PPD or 3PPD system.

5.4 All Supplier Catalogs.ext

This template can be used when user wants to create an index for items from any number of catalogs (suppliers or master). Any number of catalogs can be selected while exporting.

5.5 2PPD.ext

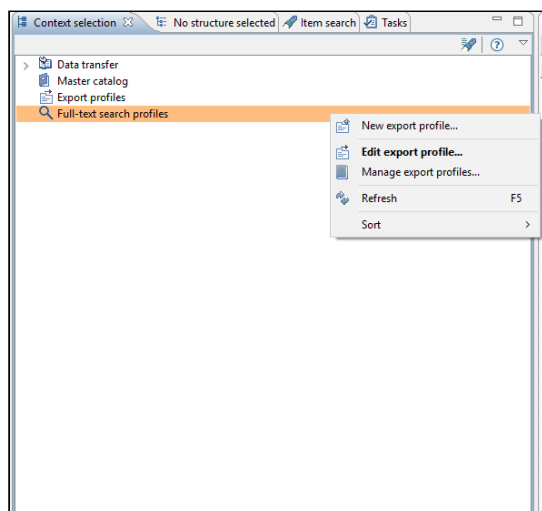
This template can be used when user wants to create an index in 2PPD system, this template provide support for new, updated, deleted and orphan records in 2PPD system, here orphan stands for records not part of hierarchy of 2PPD.

5.6 3PPD.ext

This template can be used when user wants to create an index for 3PPD system, this template provide support for new, updated, deleted and orphan records in 3PPD system, here orphan stands for records not part of hierarchy of 3PPD.

5.7 Steps to use it:

- Open Export Format Template perspective in Product 360 Desktop Client
- Select the option "Load format template from file.. "
- As per requirement, load the template into the system
- Save it on central storage
- Create a new Export profile as shown below from option "Full-text search profile"



- Now trigger the export or schedule it from the profile
- Search index should be created with the exported data

After the index creation is finished, following index will appear on web client as shown below in the Search view.

The screenshot shows the Informatica MDM web client interface. On the left is a sidebar with navigation options: Dashboard, Structures, Catalogs, Media, Tasks, and Queries. The main area displays a search result for 'Items-ori' with 9999 results. A table lists 14 items with columns: Relevance, Item no., Status, GTW, and Short description (English). The items are all 'New' and have a relevance of 5 stars. Below the table is a 'Detail view' section.

Relevance	Item no.	Status	GTW	Short description (English)
1 ★★★★★	Article_POSTMAN3333	01 New		Fugiat iusto qui perspiciatis rerum voluptas aut quia voluptates rerum.
2 ★★★★★	Article_POSTMAN3336	01 New		Dolorem nam temporibus qui sint.
3 ★★★★★	Article_POSTMAN3339	01 New		Omnis aliquid inventore earum ex fuga exercitationem.
4 ★★★★★	Article_POSTMAN3342	01 New		Eum corrupti dolore al consequatur.
5 ★★★★★	Article_POSTMAN3333	01 New		Quo sunt laborum earum autem consequatur vero quis.
6 ★★★★★	Article_POSTMAN3345	01 New		Qui dolor animi veniam quas occaecati repudiandae doloribus sit et.
7 ★★★★★	Article_POSTMAN3348	01 New		Velit expedita cupiditate iusto enim quia dolorum est aut aliquam.
8 ★★★★★	Article_POSTMAN3351	01 New		Sunt velit aut reprehenderit iure eligendi laborum rerum adipisci hic.
9 ★★★★★	Article_POSTMAN3354	01 New		In nihil rerum veniam dolores sunt omnis.
10 ★★★★★	Article_POSTMAN3357	01 New		Inventore officia et occaecati.
11 ★★★★★	Article_POSTMAN3360	01 New		Sequi qui vero alias.
12 ★★★★★	Article_POSTMAN3363	01 New		Quisquam eligendi saepe enim.
13 ★★★★★	Article_POSTMAN3366	01 New		Non vero quas.
14 ★★★★★	Article_POSTMAN3369	01 New		Eius omnis accusantium optio omnis in ratione officia vel.

6 CLAIRE Accelerator

6.1 Installation

In order to use the CLAIRE recommendation service for product data classification, you will need to have python as well as requisite machine learning libraries for training and prediction processes installed on your server.

As for recommended configurations:

- At least 4 CPU cores
- At least 10 GB of RAM, free to use by the CLAIRE recommendation service

6.1.1 Python usage on Windows

Version requirement: *Python version 3.8.5*

Unlike most Unix systems and services, Windows does not include a system supported installation of Python.

6.1.1.1 Python download and installation instructions

Conclude the following steps to make Python with its components available on your Windows machine:

Step 1: Downloading

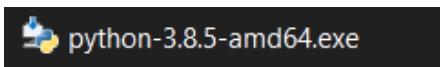
1. Open a browser window and navigate to the python for windows page
2. On the same page look for the Python 3.8.5 release in the list of the Stable Releases
3. Click on the [*Download Windows x86-64 executable installer*](#) link as shown in the screenshot bellow

▪ [Python 3.8.5 - July 20, 2020](#)

Note that Python 3.8.5 cannot be used on Windows XP or earlier.

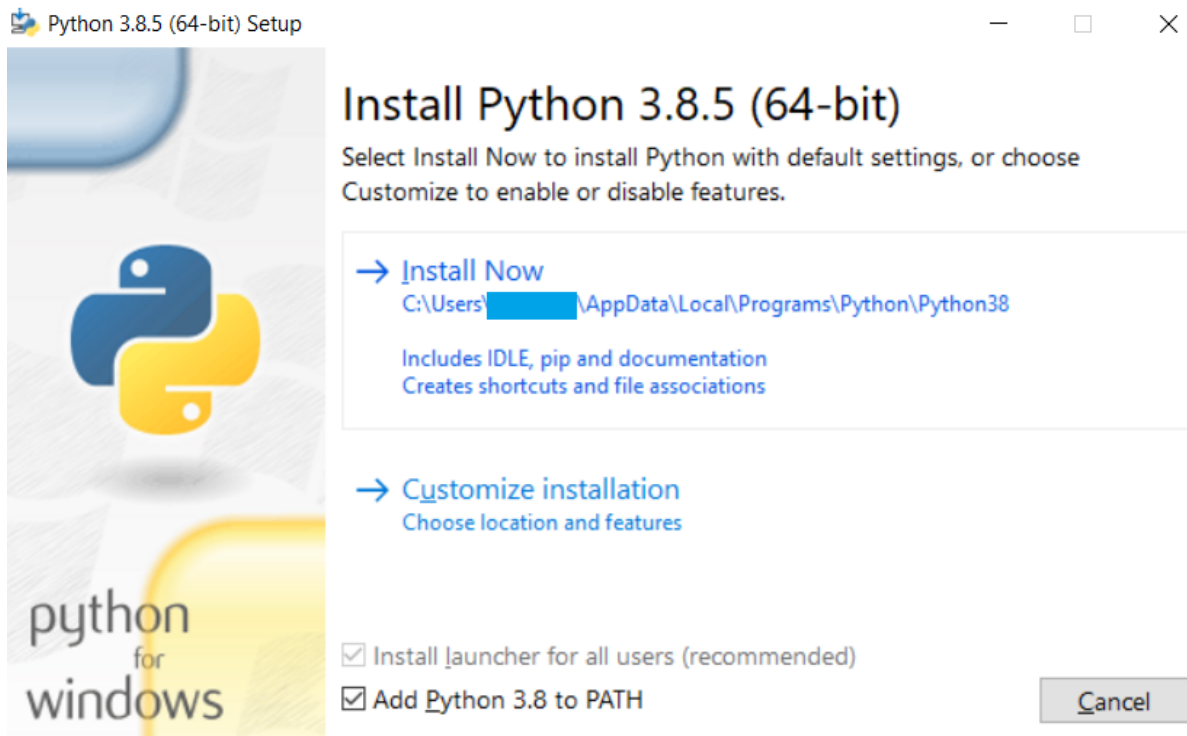
- Download [Windows help file](#)
- Download [Windows x86-64 embeddable zip file](#)
- **Download Windows x86-64 executable installer**
- Download [Windows x86-64 web-based installer](#)
- Download [Windows x86 embeddable zip file](#)
- Download [Windows x86 executable installer](#)
- Download [Windows x86 web-based installer](#)

The file named **python-3.8.5-amd64.exe** should start downloading into your standard download folder. The file should appear as



Step 2: Installing

1. Double-click the file **python-3.8.5-amd64.exe**. A **Python 3.8.5 (64-bit) Setup** pop-up window will appear.

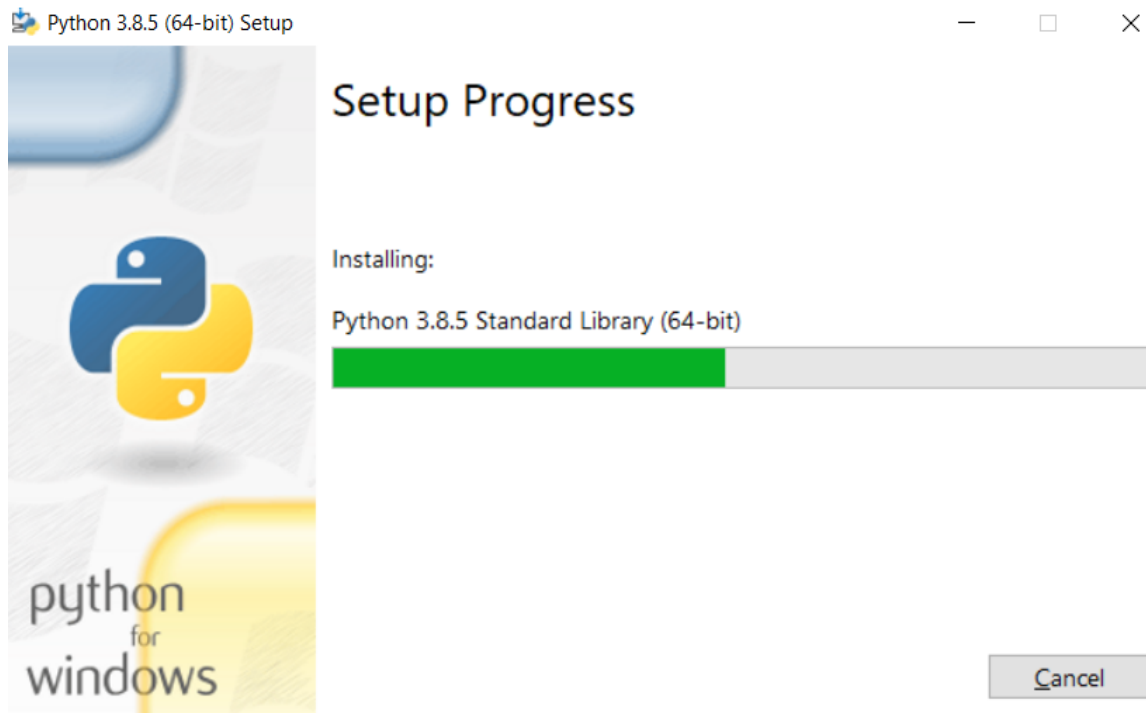


Ensure that the *Install launcher for all users (recommended)* and the *Add Python 3.8 to PATH* checkboxes at the bottom are both checked.

If the Python Installer finds an earlier version of Python installed on your computer, the *Install Now* message may instead appear as *Upgrade Now* (and the checkboxes will not appear).

2. Highlight the *Install Now* (or *Upgrade Now*) message, and then click it.

A new **Python 3.8.5 (64-bit) Setup** pop-up window will appear with a **Setup Progress** message and a progress bar.



During installation, it will show the various components it is installing and move the progress bar towards completion.

3. A new **Python 3.8.5 (64-bit) Setup** pop-up window will appear with a *Setup was successfully* message. Click the *Close* button.



Python should now be installed.

Step 3: Verifying

1. Navigate to the directory **C:\Users\%USERNAME%\AppData\Local\Programs\Python\Python38** (For the exact path see the pop-up window above in the Installing part step 1).
2. Double-click the file *python.exe*.

The following pop-up window will appear.

```
Python 3.8.5 (tags/v3.8.5:580fbb0, Jul 20 2020, 15:57:54) [MSC v.1924 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

On the first line is the python version.

6.1.1.2 Python Machine Learning libraries installation

A set of python libraries for machine learning are needed as well to make the setup complete. To install them, an installation script is provided. It's located in the CLAIRE accelerator package under *server.ai/installation*. It will automatically install all the required Machine Learning libraries with the needed version for the CLAIRE recommendation service.

All you need to do for this is to follow below steps:

- Open *File Explorer*
- Locate the installation folder with the script
- Run the script *installation-script.bat* as an admin by right clicking the batch file and selecting the *Run as administrator* option because some of the commands require administrator privileges

After you complete the steps, the batch will run each command in sequence displaying the results in the command window.

6.1.1.3 Folder creation

You have to create two folders

- **c:/informatica/ai/data** here is where the data that will be used for the training process is saved
- **c:/informatica/ai/model** here is where the trained models used for the classification process will be stored

6.1.1.4 CLAIRE server startup

Finally, you can start the CLAIRE server. A startup script *Start-CLAIRE.bat* is located in the CLAIRE accelerator package under *server.ai*. Navigate there via your Command Prompt and start the script by typing in *Start-CLAIRE.bat*.

In a few seconds, you should see the server up and running as shown in the screenshot below.

```
2021-02-22 12:55:11 DEBUG Natural language processing models [] found
2021-02-22 12:55:11 DEBUG Deep learning models [] found
2021-02-22 12:55:11 INFO Serving on 5000 ...
```

6.1.2 Python usage on Linux

Python comes preinstalled on most Linux distributions, and is available as a package on all others. To be sure use the **python --version terminal** command to check whether Python is already installed and which version you have. If Python is not installed on your Linux system, or it's not the exact version of Python 3.8.5 you have to get it installed on your machine. For that, we provide an installation script "*installation-script-linux.sh*". You can find it in the CLAIRE accelerator package under `server.ai/installation`.

The same installation will automatically install all the required Machine Learning libraries with the needed version for the CLAIRE recommendation service.



To be able to execute the installation script, you have first to make it executable by running the following command

```
chmod +x installation-script-linux.sh
```

6.1.2.1 Folder creation

You need to create the two folders for storing the data and the models **/tmp/informatica/ai/data** and **/tmp/informatica/ai/model**. The folders can be placed in a different location and you can configure the path within the `server.ai` configuration file (`config.ini`). See Chapter [Configuration and Operation](#) (see page 528) under the section *Configuration of CLAIRE recommendation service* for details.

6.1.2.2 CLAIRE server startup

As a final step, you are able to start the CLAIRE server. A startup script *Start-CLAIRE-linux.sh* is located in the CLAIRE accelerator package under `server.ai`. Navigate there via your shell and start the script.


In a few seconds, you should see the server up and running.

6.1.3 Claire server Health Check

To check if python server is up and running and the Claire application can respond to HTTP requests, we provide you an REST API endpoint that you can hit.

URL pattern	/health
Method	GET

Success Response	<ul style="list-style-type: none"> • Code: HTTP 200 • Content: OK
------------------	---

 Congrats, this should have set you up properly to work with the CLAIRE recommendation service!

6.2 Configuration and Operation

6.2.1 Configuration of CLAIRE recommendation service

A configuration file (*config.ini*) is provided. It's located in the CLAIRE accelerator package under `server.ai`. From this configuration file you are able to set parameters like authentication information, server paths and port number.

config.ini

```
[DEFAULT]
# Username and password for basic authentication for ai server. Please ensure that
# you configured Product360 CLAIRE accelerator accordingly.
user = admin
password = admin

# Folders path where to store the data and models.
upload_folder = c:/informatica/ai/data
model_path = c:/informatica/ai/model

# Python logging level. Possible values are: NOTSET, DEBUG, INFO, WARNING, ERROR,
# CRITICAL
logging_level = DEBUG

# Port number which the ai server uses
port_number = 5000
```

The parameters are mandatory and described here:

Parameter name	Description	Valid values	Default
user	Username for basic authentication against the server	<user_name>	admin
password	Password for basic authentication against the server	<user_password>	admin

Parameter name	Description	Valid values	Default
upload_folder	Path to folder where data (.csv files) used for the training process will be saved	any valid path to an existing folder	c:/informatica/ai/data
model_path	Path to folder where trained models used for the classification process will be stored.	any valid path to an existing folder	c:/informatica/ai/model
logging_level	Python logging level	CRITICAL, ERROR, WARNING, INFO, NOTSET	DEBUG
port_number	Port number used by the ai server for communication	<user_name>	5000



upload_folder and **model_path** are the same as the folders **data** and **model** you have created in the *Installation* step in the section *Folders creation*.

6.2.1.1 Preferences

Server URL

If the CLAIRE recommendation service is running on a different server than the Product360, it is possible to configure the corresponding URL in the standard plugin_customization.ini of Product360 with the following preference:

```
# The server url where the CLAIRE AI server is running.
claire.server.url = http://localhost:5000

# Usage in plugin_customization:
com.heiler.ppm.ai.core/claire.server.url = http://myserver:5000
```

User access

As preferences you set also the user and the password to access CLAIRE recommendation service.

```
# The configured user to access claire intelligence service
claire.server.user = admin

# The configured password to access claire intelligence service
claire.server.password = admin
```

```
# Usage in plugin_customization:
com.heiler.ppm.ai.core/claire.server.user = admin
com.heiler.ppm.ai.core/claire.server.password = admin
```

6.2.2 AI training

The classification of product data is supervised learning and the first step is to train a model based on existing data. To generate a model the CLAIRE accelerator leverages the power of Product 360's export. To train a model, first an export template has to be created in order to define the data for the training. This is usually the assignment to a structure group and one or more text values. In addition, the export template must configure the post processing step "*Classification training*". This post processing step will send the created file to the CLAIRE recommendation service which will eventually train a model based on the data.



6.2.2.1 Supported content languages for model generation

Machine learning models allowing to support auto classification use cases can be sensitive to the exact language being used for their training. Every individual model needs to be trained on the same content language for all records used. We have extensively tested English and German during development and received very good results with it.

Besides these two here is the overall list of languages which we expect to work with the accelerator:

- Dutch
- English
- Finnish
- French
- German
- Italian
- Norwegian
- Portuguese
- Spanish
- Swedish

i Note that any content language not listed above is likely to not work with the accelerator. Also note that we have not done full test cycles with each language listed above.

6.2.2.2 Standard export templates for AI training

With the CLAIRE accelerator package 3 pre-built export templates are provided under Resources/Export Templates and can be used as is or as template for your own training exports.

- *AI Training Items.ext*
- *AI Training Products.ext*
- *AI Training Variants.ext*

i Training data

Please ensure that there is enough training data. A training based on a few thousand records only might not bring best results. Also, the labels should be meaningful and more than just a short description like "Blue T-Shirt". Generally, the prediction results are heavily dependent on the quantity and quality of the data you feed the training with.

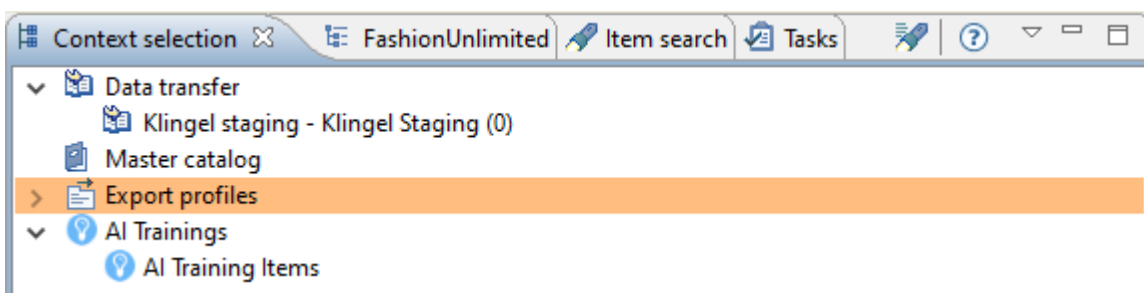
6.2.2.3 Create own export templates for AI training

It is possible to create custom export templates for AI training, e.g. if the data you want to train on is in custom fields. Some preconditions to keep in mind however are:

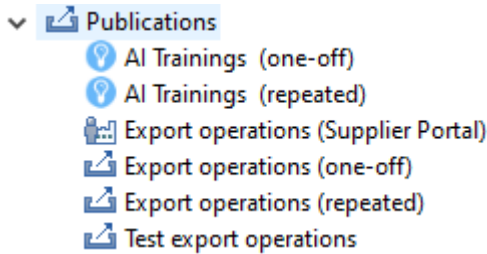
- It must be a csv file
- The purpose of the export template must be "AI Training"
- There must be a column having the header title "category". This column needs to contain the structure group assignment within the training structure.
- All label fields (can be multiple) must have the column header "label".
- The export template must have the post processing step "Classification training" attached

6.2.2.4 Create AI Training export profiles

After creating an export template or importing the examples you can create AI Trainings in the context selection view. Although the "AI trainings" are basically exports we separated them from the "normal" export to have a better user experience.



Also, in the process overview the AI trainings are separated from the "normal" exports for better process traceability



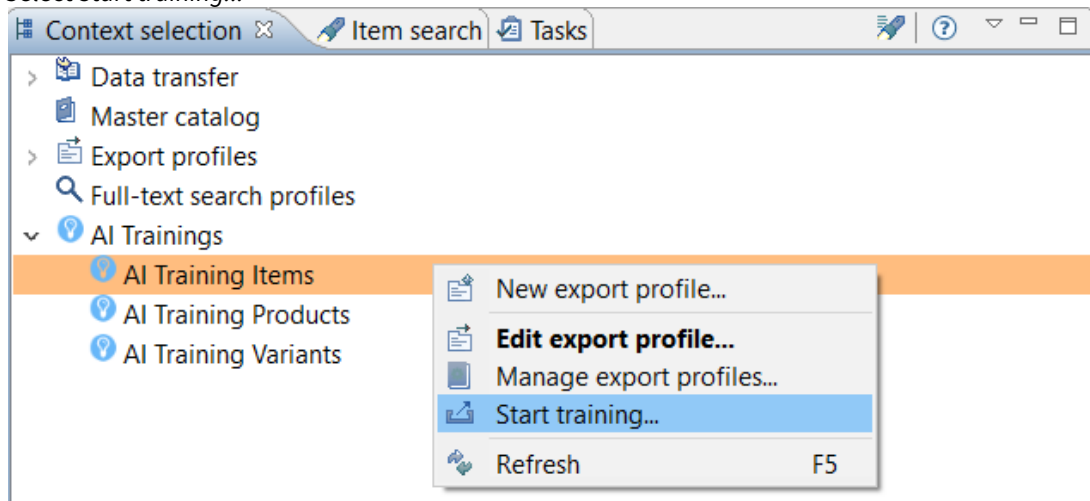
As we are using the export functionality of Product 360 you have the full power of configuring and scheduling export profiles to manage your AI trainings.

i Please note that we currently don't have any delta training. That means you have to retrain the whole model including exporting all of your data for every training.

6.2.2.5 Start AI Training

To start an AI Training, follow the steps bellow:

1. Right click on the export profile
2. Select *Start training...*



- The export parameter configuration dialog will open up. At this stage you set the export parameters for executing the training.

Export export profile - AI Training Items

Export parameter configuration
Set the export parameters for executing this export.

Data source "Item list"

Catalog:* Master catalog

Assortment:

Update assortment: Yes

Version:* Working version

Variables

Training ML approach:* Deep learning

Training duration (only deep learning): 10

Training language:* English

Training structure:* FashionUnlimited

☐ Update: before exporting

< Back Next > Finish Cancel

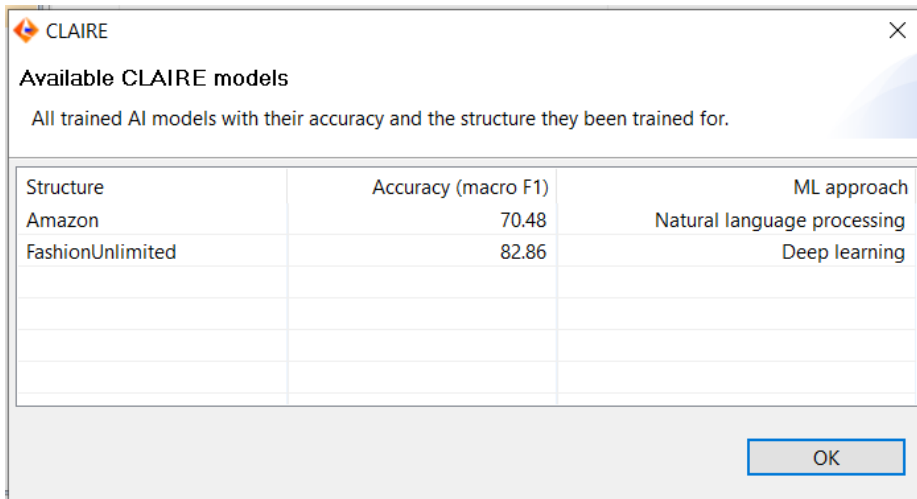
- Click on *Finish* and wait until the training is successfully completed.

Export variables

Variable name	Description
Training ML approach	The ML approach used for the training process. Possible values are "Deep learning" or "Natural language processing". See section <i>Natural language processing and deep learning explained</i> for details.
Training duration (only deep learning)	The training duration (in hours) for deep learning that should be used to train a model. By default, it's set to 10 hours which gave reasonable results in our tests. In the case of natural language processing training this parameter won't be taken into consideration and generation of a model is a lot faster.
Training language	The language of text fields exported to train model.
Training structure	The structure the training should be based on.

6.2.2.6 Display available models

There is a new entry in the "Management" menu of the Desktop UI called "Show CLAIRE models" where you can display all trained models, the algorithm they are based on, and their accuracy based on macro F1 (see chapter [Best Practices and Recommendations](#)(see page 539) under section *Model accuracy measurement* for details).



Structure	Accuracy (macro F1)	ML approach
Amazon	70.48	Natural language processing
FashionUnlimited	82.86	Deep learning

6.2.2.7 Natural language processing and deep learning explained

Deep learning:

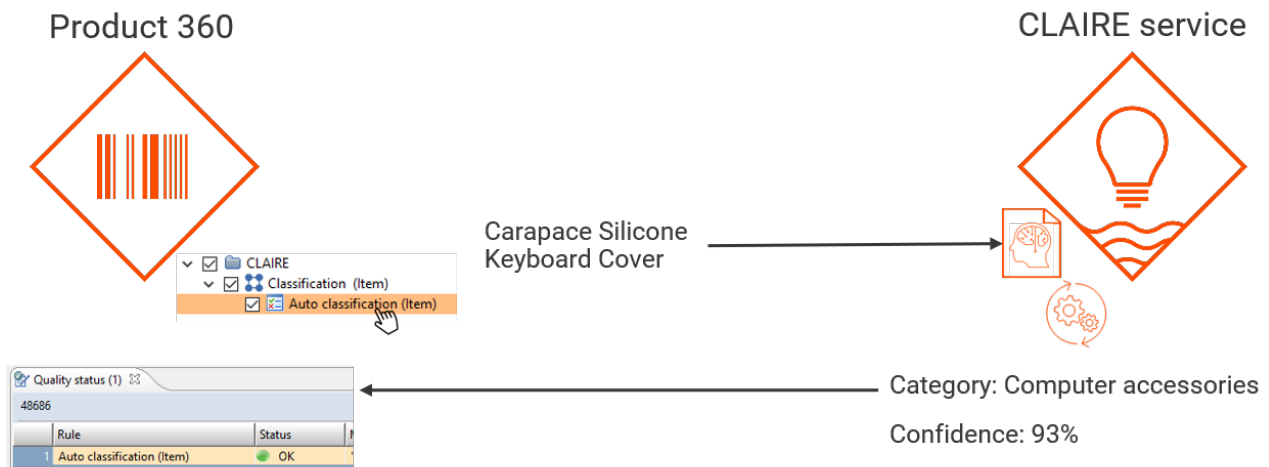
Convolutional neural network, specifically tailored for text classification tasks. Such models are based on decades of research into both AI and biology and are the technology of choice for many modern applications, including autonomous driving and speech recognition. Such networks consist of a large number of computational units that vaguely resemble some properties of neurons in human brain. Ordinarily, anyone who trains such models needs to know a lot of low-level details, to get the best performance. For this accelerator everything is however handled automatically under the hood. Tests have shown that we need around 10h of training to get a deep learning model of good accuracy.

Natural language processing:

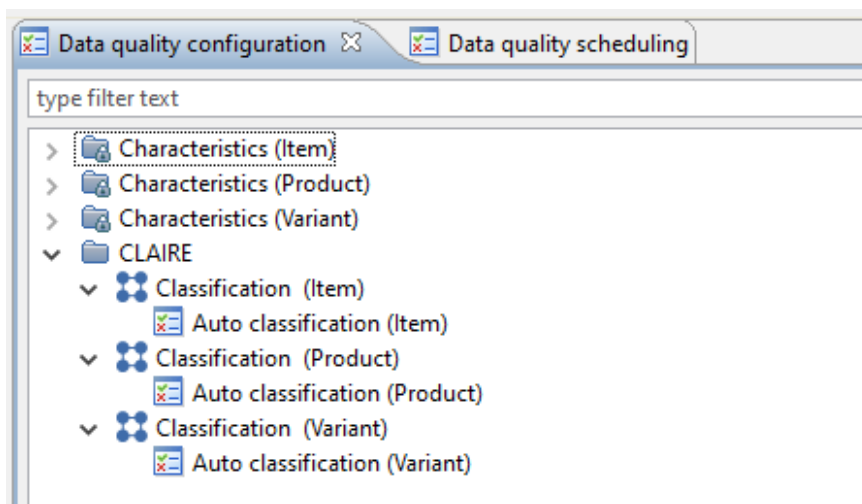
This approach produces a model rooted in classic techniques such as bag-of-words and "traditional" classifiers, such as a support vector machine or a random forest. These models are simpler, compared to deep learning, but train much faster (in our tests it took less than an hour). It might seem that such techniques are getting completely obsoleted by deep learning, but the reality is that they're still useful, although usually tend to produce models of lower accuracy.

6.2.3 Batch classification

In order to provide an auto classification in a batch process setup (for example after the import of data) the accelerator comes with the following capabilities:



After installation of the accelerator there is a new data quality category named "CLAIRE" which looks like this:



6.2.3.1 How to use the Auto classification rule configuration

i It is highly recommended to **clone** one of the existing Auto classification rule configurations and then modify the input ports according to your needs. The "default" rule configuration is just a template and should not be used as is. Even if you delete this rule configuration, category or group, it will come up on next server start again.

After the data quality rule configuration has been cloned and the parameter adjusted, it can be used for triggers or direct execution just as every other data quality rule configuration. The parameters are mandatory and described here:


Parameter	Description
Source Field	The value which will be taken as input for the prediction.
Threshold	Only recommendations with a confidence equal or above this value (in %) will be considered.
Structure identifier	Only recommendations for this structure will be considered.
Mapping mode	Set "MOVE" to replace existing mapping(s) or "ADD" to add a mapping.
ML approach	Set "DL" for using the deep learning ML approach or "NLP" for using natural language processing. See section <i>Natural language processing and deep learning explained</i> for details.
Retain existing assignments	If true, objects which already have an assignment to a group of the defined structure will be skipped.

It is possible to optionally configure a direct link within the data quality status detail tab of the Web UI which allows a user not only to spot a possible failed batch execution on item level but also to open a flex UI for fixing the classification problem right away. For that it is required to edit the file `DQNavigationDefinition.xml` which can be found in the `webdefinitions` folder of the Product 360 server. The following lines have to be added to it to link an erroneous status with a flexUI:

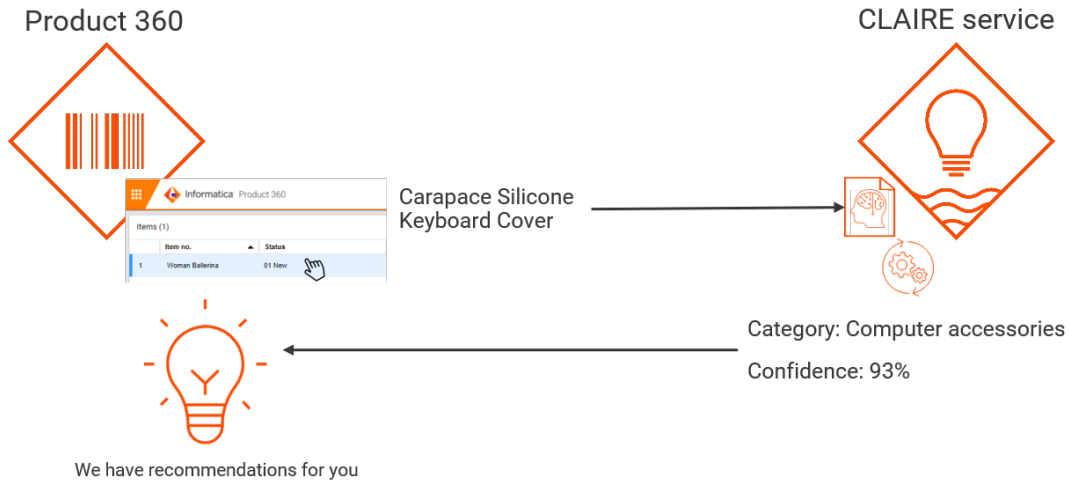
DQNavigationDefinition.xml

```
<!-- AutoClassification via CLAIRE -->
<ruleNavigation ruleName="Auto Classification Item" flexTemplateName=
"Classification with Claire" rootEntity="Article" />
```

Note that the `ruleName` property needs to be set to the actual rule name and the `flexTemplateName` to the name of the flex UI template you want to associate with the rule itself (e.g. a version of the Claire UI as seen below)

 For details on the expected performance of batch executions for auto classification please revise the chapter [Best Practices and Recommendations](#)(see page 539) under section *Recommendations for best results*.

6.2.4 Configuration of CLAIRE UI



The CLAIRE panel can be integrated into any Flex UI with a component for classification. Below you see an example of how to configure a CLAIRE panel into your flex UI definition.

```
<group identifier="Claire info">
  <layoutData>
    <parameter key="colSpan" value="1" />
    <parameter key="rowSpan" value="7" />
  </layoutData>
  <component identifier="claire full" type="claire" i18NKey="Claire" >
    <layoutData>
      <parameter key="collapsible" value="true"/>
      <parameter key="collapsed" value="false"/>
    </layoutData>
    <parameter key="context" value="classification"/>
    <parameter key="sourceField" value="ArticleLang.DescriptionLong(en)"/>
    <parameter key="algorithm" value="deeplearning"/>
    <parameter key="threshold" value="80"/>
    <parameter key="selectionThreshold" value="80"/>
  </component>
</group>
```

Parameter name	Description	Valid values	Default
context	The context or use case the Claire panel will be used for. Only recommendations for this use case will be shown. If empty or not set all recommendations will be shown.	classification	<empty>

Parameter name	Description	Valid values	Default
sourceField	The source field which will be used for prediction. The field has to be fully qualified and formatted according to the REST API syntax.	ArticleLang. DescriptionLang(de)	ArticleLang. DescriptionLang(en)
algorithm	The algorithm for classifications which will be shown. If not configured all panels for classification will be shown. See section <i>Natural language processing and deep learning explained</i> for details.	deeplearning , nlp	<all>
threshold	All predictions of CLAIRE below or equal to this threshold will not be shown in the flex UI component	0-100	80
selectionThreshold	All predictions of CLAIRE above or equal to this threshold will be automatically selected in the CLAIRE flex UI component	0-100	80

Task: CLAIRE recommendations (Item: 99)

Items (99)

	Image	Short description
1		24 HOURS Sports shoe with a 3 component midsole
2		AMY VERMONT Shirt with motive in front part
3		BABISTA Belt in classic design
4		BABISTA Fleece jacket from warming & soft berberfleece
5		BABISTA Scarf with fashionable stripe pattern
6		BABISTA Shirt with fine stripes
7		BABISTA Sliding cap with modern herringbone pattern
8		BABISTA Sweatshirt with fashionable print in collar
9		BASIL Rear wheel basket "Cento S" "Flowers"
10		BEKO Washing machine WML 61223 N
11		BELAFIT Double pack slipper in a great material mix
12		BEURER BM 35 Upper arm blood pressure monitor - fully automatic
13		BEURER MP64 Rechargeable manicure/pedicure set with 10 attachments
14		BLUE MOON Meris socks 5-pack
15		BLUE MOON sneaker socks
16		BOULEVARD DE BEAUTÉ Dazzling Eyes - Set

CLAIRE™ Recommendations

We found new recommendations for you!

Recommended classification

Structure **Fashion Unlimited**

☒ **Shirts for Women** 99%

Apply

Structure **Fashion Unlimited**

Shirts for Women

Fashion Unlimited

Fashion

Women

Shirts for Women

Textile

Laundry

Laundry Day Laundry

Laundry Shirts for Women

6.3 Best Practices and Recommendations

6.3.1 Folder structure for model maintenance

All the models you train for natural language processing and deep learning are saved in the "model" folder (as defined in the configuration file). See chapter [Configuration and Operation](#)(see page 528) under the section *Configuration of CLAIRE recommendation service*.

- If you train a model using the NLP approach, the generated model is a file named after the structure system you trained for and with the extension ".sav". For example, if you train on the structure "FashionUnlimited", the generated model will be called "FashionUnlimited.sav".
- If you train a model using the DL approach, the generated model is a folder named after the structure system you trained for. For example, if you train on the structure "FashionUnlimited", the generated model is the folder "FashionUnlimited" within your model directory.

In addition to the generated models, a JSON file is also created for each initial training. This JSON file contains information about the trained model (structure, accuracy, ML approach) and is used to provide such insights to the user via the Desktop UI (see chapter [Configuration and Operation](#)(see page 528) under the section *Display available models*). The JSON file name has the following pattern:

`<Structure_System_name> + "_" + <ML_approach> + "_metrics.json" (i.e.
FashionUnlimited_dl_metrics.json)`

In case a model has been trained on different environment it is possible to copy it into the "model" folder later on. In order to display it as available CLAIRE model on the Desktop UI, it is required to copy the corresponding generated JSON file as well. Say you train a model in a separate environment; you can copy the model file(s) as well as the JSON belonging to it into the "model" directory of your destination to make use of it properly.



If you want to delete a trained model from the "model" directory, delete its corresponding JSON file as well in order to avoid showing outdated information about a deleted model on the Desktop UI in the dialog "Available CLAIRE models".

6.3.2 Recommendations for best results

The accelerator offers two approaches to train models for product classification:

- **Natural language processing:** a pipeline based on classic algorithms from the NLP domain. Very fast to train, but in some cases, accuracy can be relatively low. These models can be quite big in size. The exact amount of space they need will depend on a particular dataset, but in our tests, some of them took as much as 30 GB. This might seem a bit excessive, but on the other hand, they should be more accurate compared to the models where the size is artificially inhibited.
- **Deep learning:** preferred approach, based on state-of-the-art research in deep learning. Such models are highly accurate, but are also time-consuming to train, especially, on big datasets.

See Chapter [Configuration and Operation](#)(see page 528) under the section *Natural language processing and deep learning explained* for details. Ordinarily, both require the user to be an expert in machine learning, but we kept all the details behind the scenes, to make the process as simple as possible. We would like to propose the following guidelines, to make sure you are getting the maximum out of this accelerator:

- **Keep the number of products in the training dataset to one million or below.** Our system and hardware requirements are tailored for datasets of this size. Also, the models we use do not need extremely big datasets to give high accuracy and going beyond one million is not likely to give a big boost.
- **For deep learning, set training time to around 10 hours.** In our tests, this timeframe resulted in models of good accuracy, and at the same time there was a probability of wasting too much time unnecessarily training much longer. Please keep in mind, that some additional time will be required for preprocessing the input data and for calculating performance metrics before the final model is available after training. For a dataset of a million records, it can take up to one hour, in addition to the time set for training. During training you can check the server window to see KPIs on the model and end the process earlier if you wish.
- **Avoid product categories that contain too few products.** The algorithms need a decent amount of data points in each category, in order to be successful. Before training even starts, we only consider categories that contain at least twenty products, the rest are discarded. It's being done because smaller categories will probably not be learned anyway, and at the same time they can harm performance on other categories.
- **Don't use inputs that are longer than 1,000 words.** All selected fields are merged into a single text input, and its size is truncated to 1,000 words. This is done to speed up training without losing accuracy.
- **Try to keep the number of product categories in 100s.** There is no hard limit on the number of product categories, but we find that a dataset of a million records can represent hundreds of categories quite well, but not thousands. Dealing with the number of classes in thousands and more usually requires much more advanced methods that are likely to require a lot of manual tuning by a data scientist, which is why we haven't further considered them in this accelerator.
- **For both natural language processing and deep learning, use small datasets when applying trained models in batch mode.** The actual speed would depend on configuration of the server, but internal tests on reasonable server hardware have shown an average performance of about 600 products/minute in our tests.

6.3.3 Model accuracy measurement

Before the training process starts, we split the input data into a training and a validation set (80/20 proportion). The model accuracy we report, is measured only on latter, to emulate how well the resulting model will perform on unseen data.

A good model performs well on two aspects:

1. If an item belongs to a certain category, the model assigns it to this category (also called *recall*).
2. If a model thinks an item belongs to a certain category, it does belong to it indeed (also called *precision*).

Both are very important. For instance, if a model just classifies anything as a watch, it will have high recall on watches, because all watches are being correctly identified. However, its precision will be low, as there will be many items from other categories incorrectly identified as watches.

This is why we use a measure called F1, which combines both recall and precision (this is a well-established and well-known metric used in many applications of machine learning). We calculate it for every product class in validation set separately, and then report the average of these scores (the technical term for that is *macro-F1*). It can take values between zero (bad) and one (good). Bigger values represent higher precision and recall across all product categories.

A word of caution: it is not classification accuracy and can't be compared with it directly!

Here are some guidelines on how to interpret F1 scores:

Value interval	Remarks
0.80-1.00	These values are quite rare, and usually can be achieved with a small number of classes only in the model (tens, not hundreds).
0.60-0.80	In our tests, these models gave good accuracy across product categories, but occasionally performed a bit less successfully on a small number of categories.
0.40-0.60	Perform well on the majority of product categories.
0.00-0.40	We wouldn't recommend using such models, as their performance can be unreliable. If you're seeing such scores, try reducing the number of classes and/or supplying more training data.

7 Swagger UI

7.1 Purpose

The purpose of this document is to educate the user on the usage of the Product 360 Swagger UI accelerator.

The Swagger UI is a REST API client for the visual interaction with Informatica MDM - Product 360 Service APIs, specifically the List and Management API components.

This document covers all the aspects of the application setup, right from installation to adoption. This guide assumes basic understanding of Product 360 and corresponding REST functionality.

For more information on Product 360 and REST functionality, please refer to the Service API documentation packaged with the application.

7.2 Installation

7.2.1 Prerequisites

The users using this application are expected to have at least a basic understanding of Informatica MDM - Product 360 and corresponding Service API functionality.

The setup/application prerequisites include:

- Oracle Java SE 8 or above or Azul OpenJDK 8 and above
- Browser (Chrome, Firefox or Microsoft Edge)
- Informatica MDM - Product 360 Server 10.x or above

7.2.2 Terms

Below are the quick references to terms that the user may come across at multiple points:

- **Swagger** - Swagger is an open-source software set of tools to design, build, document, and use RESTful web services, developed by SmartBear Software.
- **Metadata Engine** - Java based dynamic REST API builder for Product 360 functionalities of List API and Meta API components using the repository data.
- **REST** - REST, or **RE**presentational **St**ate **T**ransfer, is an architectural style for providing standards between computer systems on the web, making it easier for systems to communicate with each other.

7.2.3 Download package

The Informatica MDM - Product 360 Swagger UI accelerator can be found within the **PIM_10.1.0.00.00_Accelerators.zip**.

Once downloaded, extract the zip to a folder. Preferably Informatica folder for ease of access. The extracted folder would look like below:

logs	File folder
service installation	File folder
static_out	File folder
application.properties	PROPERTIES File
log4j2.xml	XML File
spring-boot-swagger2-0.0.1-SNAP...	WAR File
start_swagger_ui.bat	Windows Batch File
start_swagger_ui.sh	SH File

7.2.4 Operation as a Windows Service

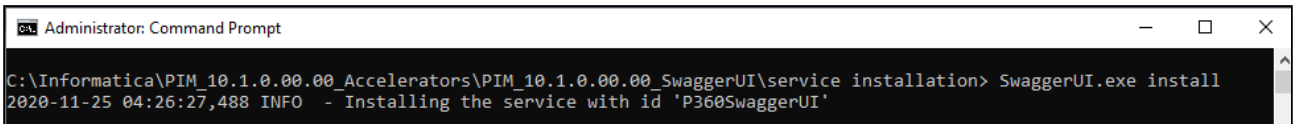
Navigate to the *SwaggerUI/service_installation* folder

logs	File folder
static_out	File folder
startup_swaggerui_service.bat	Windows Batch File
startup_swaggerui_service.sh	SH File
SwaggerUI.exe	Application
SwaggerUI.xml	XML File

To install Swagger UI as a Windows service, open command prompt in "Administrator Mode" and run the below command from extracted folder directory

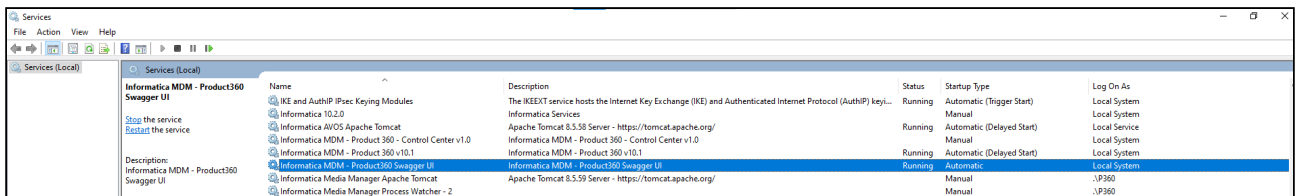
```
SwaggerUI.exe install
```

 Ensure Java and JDK/JRE are added to **PATH** / **JAVA_PATH** variables in the Windows Environment



```
Administrator: Command Prompt
C:\Informatica\PIM_10.1.0.00.00_Accelerators\PIM_10.1.0.00.00_SwaggerUI\service installation> SwaggerUI.exe install
2020-11-25 04:26:27,488 INFO - Installing the service with id 'P360SwaggerUI'
```

Open "Services" app and start the **Informatica MDM – Product 360 Swagger UI** service












To uninstall the Swagger UI service, open command prompt in "Administrator Mode" and run the below command from extracted folder directory

```
SwaggerUI.exe uninstall
```

7.2.5 Operation in Standalone Mode

To run the Swagger UI as a standalone utility, navigate to the *SwaggerUI* folder and launch the application by executing the `start_swagger_ui.bat` / `start_swagger_ui.sh` scripts for Windows or Linux respectively. The application would open the login page of Product 360 Swagger UI in the default browser.

 Ensure Java and JDK/JRE are added to **PATH** / **JAVA_PATH** variables in your Windows/LINUX Environment

 spring-boot-swagger2-0.0.1-SNAPSHOT....	WAR File
 start_swagger_ui.bat	Windows Batch File
 start_swagger_ui.sh	SH File
 application.properties	PROPERTIES File
 log4j2.xml	XML File
 logs	File folder
 service installation	File folder
 static_out	File folder

7.3 Configuration and Operation

7.3.1 Configuration

The following application configurations are provided within `application.properties`, for adjusting the operation of Swagger UI

application.properties

```
# Title for displaying within the Swagger UI context
application.title=Informatica MDM - Product 360 Swagger UI

# Context settings for Swagger UI URL
server.contextPath=/swagger
server.port=9080

# Specify the maximum number of threads used for building the OpenAPI specification
YAML file.
# Allowed value is an integer greater than 0.
server.maxThreads = 10

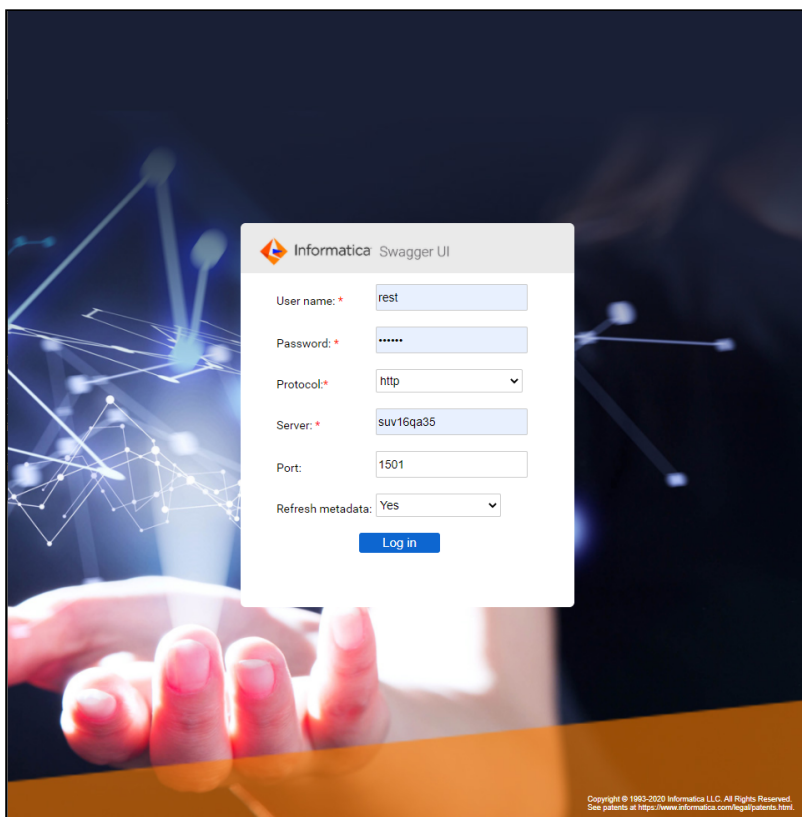
# The maximum time for the user session in minutes
server.maxSession= 300

spring.mvc.view.prefix=/WEB-INF/views/
spring.mvc.view.suffix=.jsp
spring.output.ansi.enabled=ALWAYS
application.formatted-version=1.0
```

7.3.2 Operation

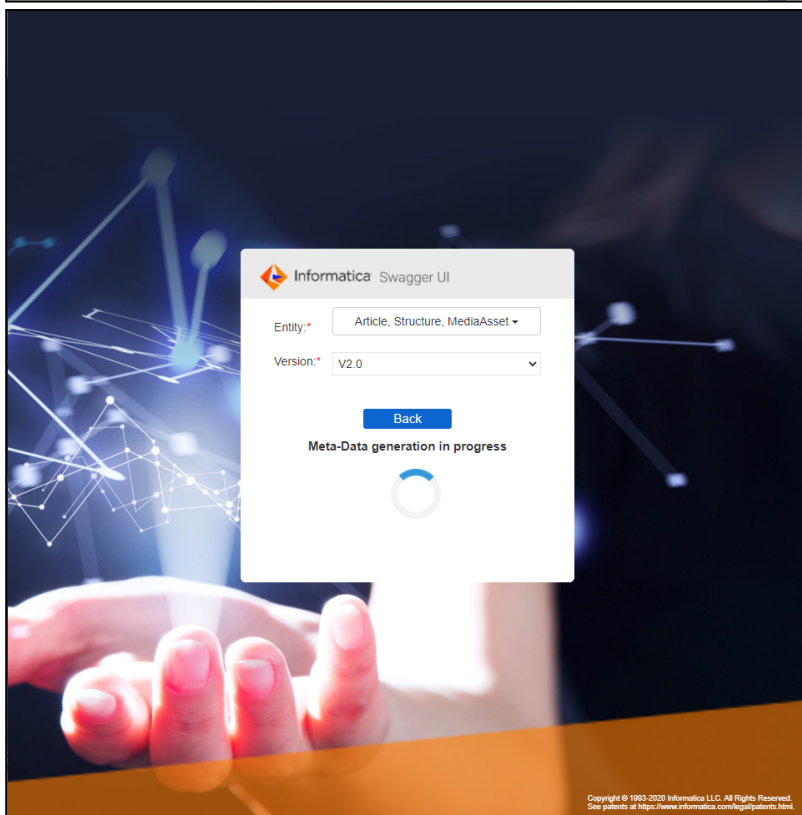
The Swagger UI landing page, the login window, should be presented in the default browser with URL "http://localhost:9080/swagger/swagger-ui.html". This URL reflects the default properties set in the configurations. This might be different based on your setup.

Enter the REST API User and Product 360 Server credentials received from the admin, into the login window. For the first time, select "Yes" from the "Refresh Metadata" drop-down and click login. At this point the application fetches the latest repository information and builds a list of entities available within the specific Product 360 installation. In case of no interim changes in the repository, for future logins this step can be skipped by choosing "No" under "Refresh metadata". However, with any changes to the entity information in the repository this metadata refresh action should be performed again.

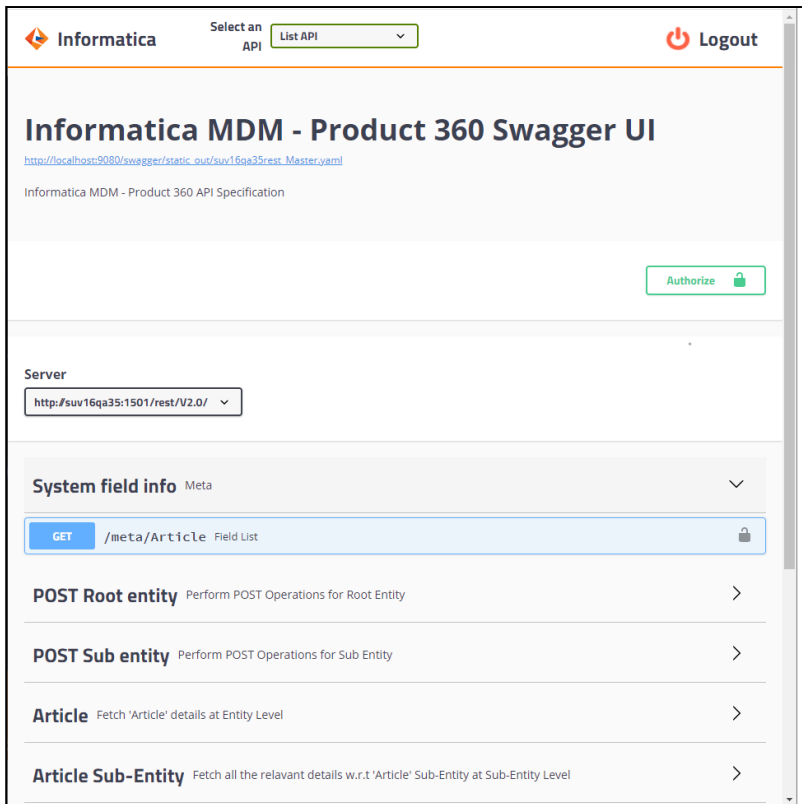


Once the metadata is refreshed, a page with the supported entities will be presented. Select the entity types, that you would like to operate upon using the List API, and click "Generate". This action initiates the generation of the custom repository specific OpenAPI specification for the supported Service API.

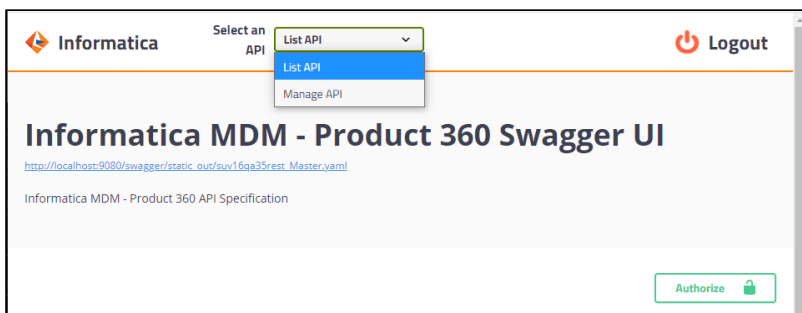
Depending upon the number of entities / sub-entities in the repository and the network overhead, the metadata generation and the corresponding OpenAPI specification generation might take a few minutes. As mentioned earlier, this step would have to be performed only whenever there is any change in the metadata information.



Once the OpenAPI specification is generated, the home page of Product 360 Swagger UI, as shown below, should appear.



Select one of the API specifications from the drop-down list. Once a selection is made the corresponding OpenAPI YAML specification file can be downloaded, if needed for client REST API implementations, from the link appearing under the Informatica logo.



Before executing any query, you must be authorized. Click on the Authorize button on the home page and enter your user credentials. Once successfully authorized you can execute the required operations. Later pages show an example for the execution of a GET operation for an entity.

Available authorizations

httpBasic (http, Basic)

Username:

rest

Password:

.....

Close

Authorize

7.3.2.1 Example : Fetch 'Article' details at Entity Level, by Items in a catalog

On the Swagger UI page is a list of entity and sub-entity panels corresponding to the selection made earlier, while generating the OpenAPI specification. Upon clicking any of the entity / sub-entity panels, the server stubs to the respective API endpoints and permitted REST operations would be displayed.

For the current example we choose Article > GET /list/Article/byCatalog

Server

http://suv16qa39:1501/rest/V2.0/

System field info

Meta

>

POST Root entity

Perform POST Operations for Root Entity

>

POST Sub entity

Perform POST Operations for Sub Entity

>

Article

Fetch 'Article' details at Entity Level

>

Article Sub-Entity

Fetch all the relevant details w.r.t 'Article' Sub-Entity at Sub-Entity Level

>

Structure

Fetch 'Structure' details at Entity Level

>

Structure Sub-Entity

Fetch all the relevant details w.r.t 'Structure' Sub-Entity at Sub-Entity Level

>

MediaAsset

Fetch 'MediaAsset' details at Entity Level

>

MediaAsset Sub-Entity

Fetch all the relevant details w.r.t 'MediaAsset' Sub-Entity at Sub-Entity Level

>

Article

Fetch 'Article' details at Entity Level

>

GET

/list/Article/byAssortment

Items in assortment

🔒

GET

/list/Article/byCatalog

Items in a catalog

🔒

Returns all items in the specified catalog. The master catalog (default) or any supplier catalog may be specified here.

Parameters

Try it out

On selecting the API endpoint, supported parameters and their descriptions are displayed. To set the parameters and execute, click the "Try it out" button. A set of text boxes, drop-downs and multi-select fields against each parameter would now be enabled

GET /list/Article/byCatalog Items in a catalog

Returns all items in the specified catalog. The master catalog (default) or any supplier catalog may be specified here.

Parameters

Name	Description
catalog string (query)	The catalog for which the items are to be determined
revision string (query)	The version in which the items must exist
compareRevision string (query)	The version against which all items that have changed have to be displayed.
dataQualityStatus string (query)	The data quality status to be used as the filter.
characteristicValueFilter string (query)	The characteristic value filter query
channel string (query)	The channel to be used as the filter.
ruleConfiguration string (query)	The validation rule to be used as the filter.
fields array (string) (query)	Fields to be displayed in the result

Cancel

GET /list/Article/byCatalog Items in a catalog

Returns all items in the specified catalog. The master catalog (default) or any supplier catalog may be specified here.

Parameters

Name	Description
catalog string (query)	The catalog for which the items are to be determined
revision string (query)	The version in which the items must exist
compareRevision string (query)	The version against which all items that have changed have to be displayed.
dataQualityStatus string (query)	The data quality status to be used as the filter.
characteristicValueFilter string (query)	The characteristic value filter query

Try it out

From the listed options, API parameters can be set by selecting fields from dynamically loaded drop-down lists (use Ctrl+Select / Shift+Select, for selecting multiple options) or by entering text based on the parameter type. In the example below, Catalog and display fields parameters are chosen from a drop-down list and parameters such as pageSize are set explicitly using the text box against it. Obligatory required parameters are indicated with a *required against the parameter.

Name	Description
catalog string (query)	The catalog for which the items are to be determined
revision string (query)	Must exist
compareRevision	Items that have changed have to be displayed.

fields	Fields to be displayed in the result
array [string] (query)	<div> Article.AclProxy Article.CatalogProxy Article.ContentUnit Article.CurrentStatus </div>

Once the necessary parameters are set, clicking “Execute” would generate the Curl and Request URLs, that can be used directly in downstream REST API client applications and workflows. The selected query is also executed and the response for the query is shown in the "Response body" section in XML/JSON format based on the selection in the "Accepted Header" parameter.

Execute

Clear

Responses

Curl

```
curl -X GET "http://suu16qa35:1501/rest/V2.0/List/Article/byCatalog?catalog=Master%20catalog&fields=Article.AclFlag,Article.AclProxy,Article.CatalogProxy&pageSize=10" -H "accept: application/xml" -H "Authorization: Basic cmVzdDpoZWlsZXI="
```


Request URL

```
http://suu16qa35:1501/rest/V2.0/List/Article/byCatalog?catalog=Master%20catalog&fields=Article.AclFlag,Article.AclProxy,Article.CatalogProxy&pageSize=10
```


Server response

Code	Details
200	<div>Response body</div> <pre><?xml version="1.0" encoding="UTF-8" standalone="yes"?> <entityItemTable> <cacheId>no-cache</cacheId> <entityIdentifier>Article</entityIdentifier> <totalSize>1248</totalSize> <startIndex>0</startIndex> <pageSize>10</pageSize> <rowCount>10</rowCount> <columnCount>0</columnCount> <columns> <rows> <row> <object> <id>1301</id> <entityId>1000</entityId> </object> <values> <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xs="http://www.w3.org/2001/XMLSchema" xsi:type="xs:string"> </value> <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xs="http://www.w3.org/2001/XMLSchema" xsi:type="xs:string"> </value> <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="entityItemReference"> <id>1</id> <entityId>2900</entityId> </value> </values> </row> </rows> </columns> </entityItemTable></pre>

Other REST API operations can be executed in a similar fashion as mentioned above.

7.4 Best Practices, Recommendations and Known Issues

For the optimal usage of Swagger UI, the following recommendations are made:

- The installation of the Swagger UI as a service on the same node as that of the Product 360 Server, prevents any latency on account on network and thereby would lead to quicker generation of metadata and OpenAPI specification.
- In case of repositories with complex metadata and large number of entities/sub-entities the following optimizations, through the Swagger UI configuration settings, could ensure timely generation of the specification:
 - Increasing the JVM heap size, recommended default value is 512mb.
 - Increasing the number of threads, recommended default value is 10 threads.

Additionally, the following limitations are known for the current version of Swagger UI:

- It currently supports only the List API and Management API components within Product 360
- Channel Entity as part of Product 360 10.1, is not yet supported

8 Micrometer Metrics and Dashboard

With the inbuilt Micrometer instrumentation façade, Product 360 10.1 delivers over 150 system and application-specific dimensional metrics for actionable health insights and reporting of the environment. With vendor-neutral meters, it is now possible to independently provision an application performance monitoring system of your choice. Additionally, Product 360 10.1 ships with an accelerator that includes predefined Elastic Kibana based reference dashboards providing an overview of all key metrics for application monitoring - ready to use and adopt as needed.

The purpose of this document is to guide the user regarding the installation, configuration and usage of the Elastic Kibana based reference dashboards.

8.1 Installation, Configuration and Operation



8.1.1 Installation





8.1.1.1 Accelerator package

The application monitoring dashboards are available as out-of-the-box visualizations, in JSON file format, which can be imported directly through the Kibana UI.

The JSON files and reference configurations can be found in the within the **PIM_10.1.0.00.00_MicrometerDashboard** directory in the **PIM_10.1.0.00.00_Accelerators.zip**

Once downloaded, extract the zip to a folder. Preferably Informatica folder for ease of access. The extracted folder would look as below:

<input type="checkbox"/> Name	Size
 P360 Micrometer Kibana Dashboard Config	
 Recommended-log4j2-Micrometer.xml	4 KB

<input type="checkbox"/> Name	Size	Type
 p360_advanced_settings.ndjson	1 KB	NDJSON File
 p360_kibana_dashboard.ndjson	712 KB	NDJSON File
 p360_kibana_index.ndjson	16 KB	NDJSON File
 p360_kibana_visualization.ndjson	585 KB	NDJSON File

8.1.2 Configuration

8.1.2.1 ElasticSearch Configuration

The ElasticSearch configuration has to be done in the **micrometer.properties** file of the corresponding application server. Scroll down to the section for Elastic search and ensure the settings as shown below

micrometer.properties


```
#####
### Elastic settings                                     ###
#####
elastic.enabled = true
# Ensure that access credentials for elastic search host are added correctly
elastic.host =
elastic.userName =
elastic.password =
#Do not change the name of the index, otherwise the provided Kibana visualisations
#will also need to be adjusted as the index name is configured there!
elastic.index = p360_metrics
elastic.indexDateFormat = yyyy-MM
elastic.timestampFieldName = @timestamp
elastic.autoCreateIndex = true
elastic.step = 1m

# Optional parameters
#elastic.pipeline =
#elastic.indexDateSeparator =
```

```
#elastic.connectTimeout      =
#elastic.readTimeout         =
#elastic.batchSize           =
```

8.1.2.2 Metrics Configuration

Make sure to set the proper `system.name` property in each `server.properties` file as well since this property will be used for the `appStack` tag. The system name will also be used as a prefix for `fulltextsearch` index name.

 Blanks will be replaced with `_`. Best practice: use `0-9A-Za-z.-`

server.properties

```
#####
### System Settings                                     ###
#####
# Specifies the name of the system, e.g. Test System /Productive System / Demo / Poad
etc.
# The system name will also be used as a prefix for fulltextsearch index name.
# Blanks will be replaced with _. Best practice: use 0-9A-Za-z.-
system.name =
```

The Metrics configuration is possible in the **log4j2.xml** of the corresponding application server. Scroll to the "Metrics" section and replace the existing configuration with the settings below. Micrometer metrics can be enabled completely or for specific sub categories.

For example enabling **metrics.databaseTables.job** sub category and disabling the parent **metrics.databaseTables** metrics category, will disable all database table metrics except for database table job metrics.

 Use "ERROR" to disable and "INFO" to enable metrics.

log4j2.xml

```
1  <!-- ===== -->
2  <!-- Metrics      -->
3  <!-- ===== -->
4
5  <!-- Micrometer metrics can be enabled completely or for specific sub
categories:
6      Use "ERROR" to disable and "INFO" to enable metrics.
7          - "metrics.cache" activates Cache metrics
8          - "metrics.communication" activates Communication Framework metrics
9          - "metrics.databaseTables" activates database tables metrics:
10         - "metrics.databaseTables.masterCatalog"
```

```

11         - "metrics.databaseTables.supplierCatalogs"
12         - "metrics.databaseTables.structure"
13         - "metrics.databaseTables.otherEntities"
14         - "metrics.databaseTables.job"
15         - "metrics.databaseTables.meta"
16     - "metrics.dataGraph" activates DataGraph metrics
17     - "metrics.dataQuality" activates DataQuality metrics.
18     - "metrics.detailModel" activates DetailModel metrics
19     - "metrics.hibernate" activates Hibernate metrics
20         - "metrics.hibernate.query" activates Hibernate Query metrics
21     - "metrics.http" activates Http Request metrics.
22         - "metrics.http.response" activates Http Response metrics.
23     - "metrics.jetty" activates Jetty metrics
24     - "metrics.log4j2" activates Log4j2 metrics
25     - "metrics.login" activates Login metrics.
26         - "metrics.login.user" activates Login user metrics.
27     - "metrics.mediator" activates Mediator metrics
28         - "metrics.mediator.submediator" activates Sub-Mediator metrics
29     - "metrics.persistenceManager" activates PersistenceManager metrics
30     - "metrics.restClient" activates REST client metrics
31     - "metrics.auditTrail" activates Audit trail metrics
32     - "metrics.import" activates Import metrics
33         - "metrics.import.persistence" activates Import persistence
metrics
34         - "metrics.import.storage" activates Import storage metrics
35         - "metrics.triggerService" activates Trigger Service metrics
36     -->
37
38     <Logger name="metrics" level="INFO" additivity="false" />
39
40     <!-- Following metrics are disabled by default for performance reasons,
41     enable only for debugging -->
42     <!--Cache, Communication, http, hibernate -->
43     <Logger name="metrics.cache" level="ERROR" additivity="false" />
44     <Logger name="metrics.communication" level="ERROR" additivity="false" />
45     <Logger name="metrics.http" level="ERROR" additivity="false" />
46     <Logger name="metrics.hibernate" level="ERROR" additivity="false" />
47
48     <!-- Database Tables metrics are disabled by default, Except for Job, for
49     performance reasons, enable only for debugging-->
50     <Logger name="metrics.databaseTables" level="INFO" additivity="false" />
51     <Logger name="metrics.databaseTables.job" level="INFO" additivity="false"
52     />
53     <Logger name="metrics.databaseTables.masterCatalog" level="ERROR"
54     additivity="false" />
55     <Logger name="metrics.databaseTables.supplierCatalogs" level="ERROR"
56     additivity="false" />
57     <Logger name="metrics.databaseTables.structure" level="ERROR" additivity="
58     false" />
59     <Logger name="metrics.databaseTables.otherEntities" level="ERROR"
60     additivity="false" />
61     <Logger name="metrics.databaseTables.meta" level="ERROR" additivity="false
62     " />

```

```

55
56 <!-- Sub mediator metrics are disabled by default for performance reasons,
57    enable only for debugging-->
57 <Logger name="metrics.mediator" level="INFO" additivity="false" />
58 <Logger name="metrics.mediator.submediator" level="ERROR" additivity="false" />

```


8.1.2.3 Import of Kibana Saved Objects


Once the above mentioned Product 360 server configurations are complete, the Kibana JSON objects for settings, index pattern, visualisations and dashboards can be imported.

For this open the **Kibana > Management > Saved Objects** page and Import the JSON objects "P360 Micrometer Kibana Dashboard Config" directory, in the following order:

1. **p360_advanced_settings.ndjson**: Optimal settings for performant aggregation queries
2. **p360_kibana_index.ndjson**: The Kibana index pattern for "p360-metrics-idx" mapping the metrics files to the Product 360 metrics index
3. **p360_kibana_visualization.ndjson**: The individual Kibana metric visualisations
4. **p360_kibana_dashboard.ndjson**: The 3 Product 360 application and system monitoring dashboards

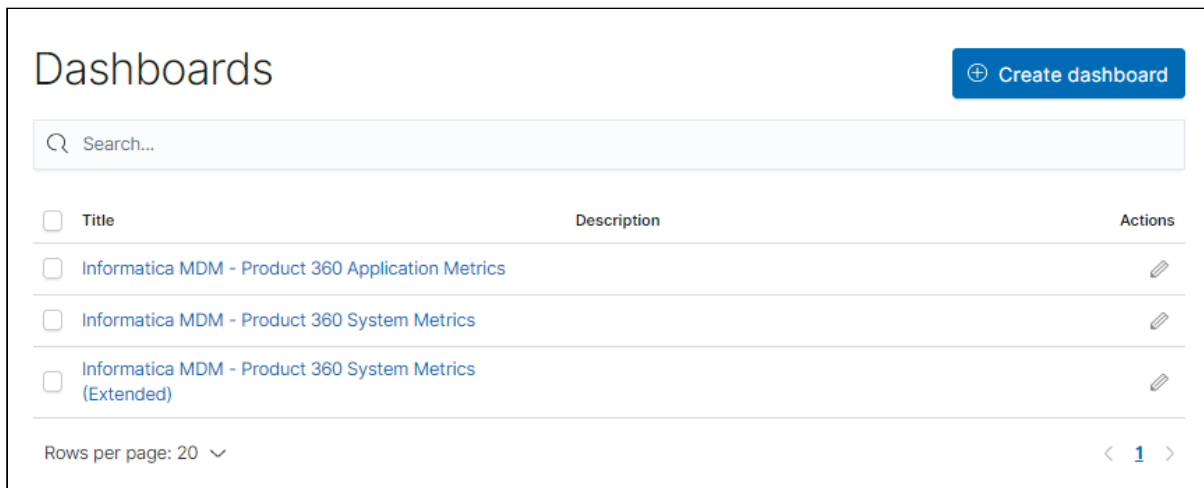
Once the above objects are successfully imported, ensure the following settings are enabled in the **Kibana > Management > Advanced Settings**

-  Default index : p360-metrics-idx
- Index pattern placeholder : p360_metrics-*
- Store URLs in session storage : On
- Maximum buckets : 20000

-  Please note that from version Elastic Kibana 7.7 onwards, you might have to create a dedicated space for p360 metrics first before importing the saved objects. The imported settings, index pattern, visualisations and dashboards should reside in this space.

8.2 Dashboards

The Product 360 Kibana Dashboards should be available under **Kibana > Dashboards** as shown below :



The Product 360 Micrometer based monitoring is achieved through the following three Kibana Dashboards:


Dashboard name	Purpose
Informatica MDM - Product 360 Application Metrics	Product 360 specific application metrics that a Business administrator could use to monitor and track relevant statistics
Informatica MDM - Product 360 System Metrics	Product 360 server metrics that a System administrator could use to monitor and track relevant statistics
Informatica MDM - Product 360 System Metrics (Extended)	Low level Product 360 server metrics that a System administrator could use to monitor and track relevant statistics

8.2.1 Global Filters

In order to use the dashboards efficiently the following global filters are provided that will filter all metrics against the selection(s) of the filter.

8.2.1.1 Application Stack

The application stack filter allows the use of multiple P360 installations within the same Elasticsearch index. For example a single index can store the metrics for DEV, TEST, QA and PROD installations and the application stack filter allows easy switching between them.


 Please note that all metrics can be filtered against the application stack.

Metric Tag

appStack

8.2.1.2 Application Server

The application server filter allows the use of multiple P360 application servers within the same Elasticsearch index. All metrics will be filtered against the selection of this filter. For example a single index can store the metrics for a multi-server installation of P360, consisting of pim-server1 and pim-server2, and the application server filter allows easy switching between them.

 Please note that all metrics can be filtered against the application stack.


Metric Tag

appId

8.2.1.3 Entity Type

The entity type filter allows filtering (drill-down) of metrics regarding their corresponding repository entity types.

For example it is possible to drill-down into all "item" (ArticleType) metrics.


 Please note that not all metrics are related to repository entity types and won't show any data while the entity type filter has been selected.

Metric Tag

entityType

8.2.1.4 Entity

The entity filter allows filtering (drill-down) of metrics regarding their corresponding repository entities. For example it is possible to drill-down into all "item" (Article) metrics.


 Please note that not all metrics are related to repository entities and won't show any data while the entity filter has been selected.

Metric Tag

entity

8.2.1.5 Database

The database filter allows filtering (drill-down) of metrics regarding any kind of database activity. For example it is possible to show all active connections for a specific database.


 Please note that not all metrics are related to the database and won't show any data while the database has been selected.

Metric Tag

pool

8.2.1.6 Message Type

The message type filters allows filtering (drill-down) of the communication framework for specific messages (every message in the communication framework has a specific message type). For example it is possible to show the amount and duration of rich client which request data from the database.

 Please note that only the communication framework metrics are message type specific, all other metrics won't show any data while the message type has been selected.

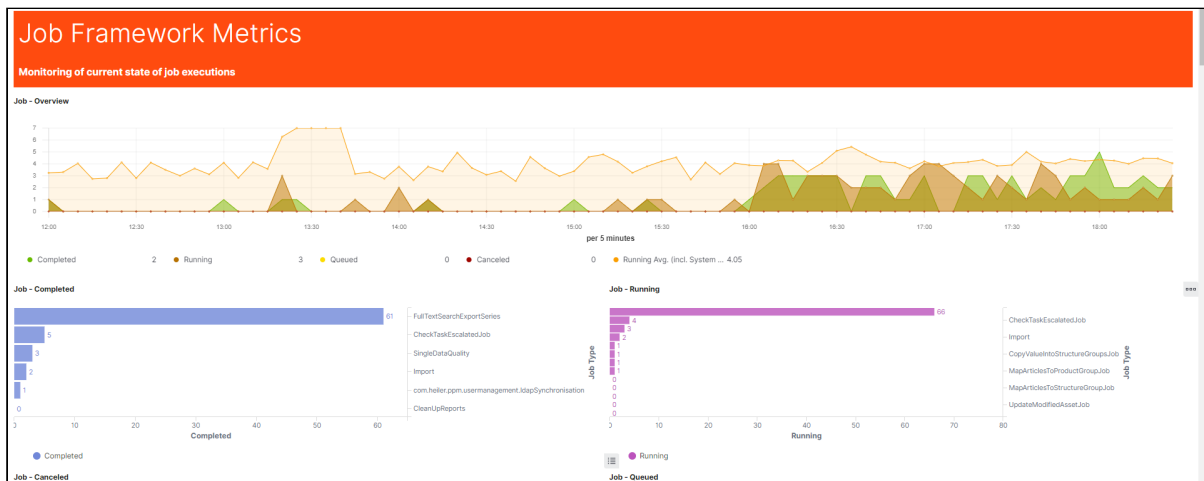
Metric Tag

messageType

8.2.2 Application Metrics Dashboard

8.2.2.1 Job Framework Metrics

Monitoring of current state of job executions.



Metric Name	Metric Key	Description
Canceled	job_canceled	The amount of canceled jobs
Completed	job_completed	The amount of completed jobs
Queued	job_queued	The amount of queued jobs
Running	job_running	The amount of running jobs

8.2.2.2 Data Quality Metrics

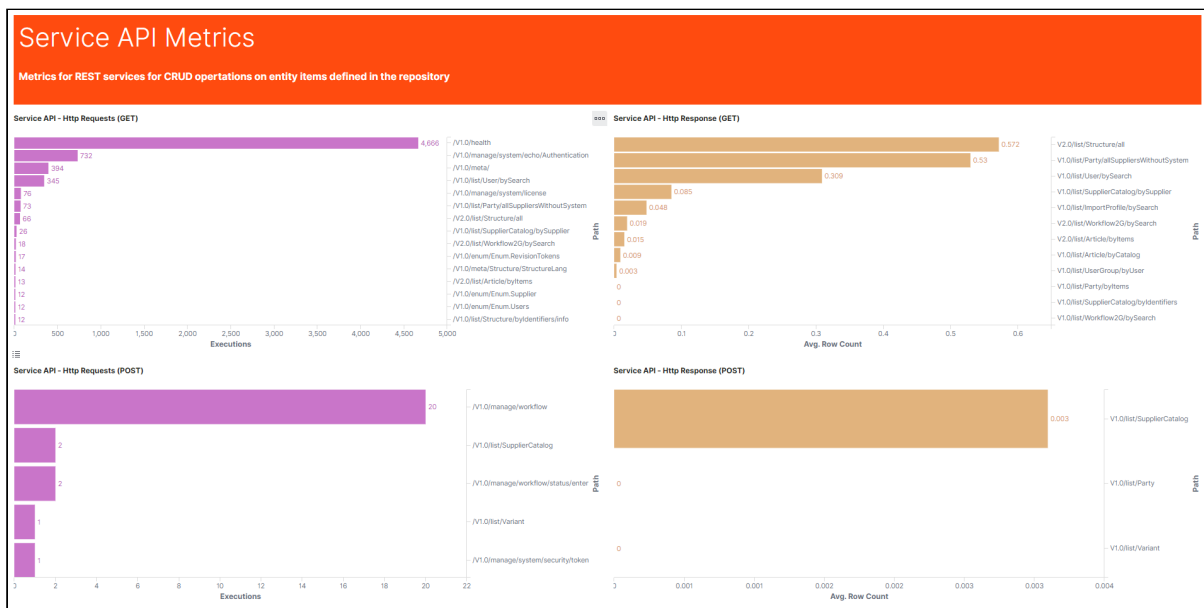
Statistics to monitor the various DQ mapping and rule executions.



Metric Name	Metric Key	Description
Mapping Execution	dataQuality_executing	The amount of Data Quality rules currently executing by the system
Mapping Executor Creation	dataQuality_executor_creation	The amount of Data Quality Mapping Executors created by the system
Mapping Execution Wait	dataQuality_waiting	The amount of time the Mapping Executors had to wait for a free execution slot

8.2.2.3 Service API Metrics

Metrics for REST services for CRUD operations on entity items defined in the repository.



Metric Name	Metric Key	Description
HTTP Requests	httpRequest	The amount of HTTP requests
HTTP Responses	httpResponse_list	The amount of objects in the HTTP response

8.2.2.4 Persistence Metrics

Metrics for various persistence resources and service groups.

The global persistence filters include both **EntityType** and **Entity** from the repository.

Persistence Metrics

Metrics for various persistence resources and service groups

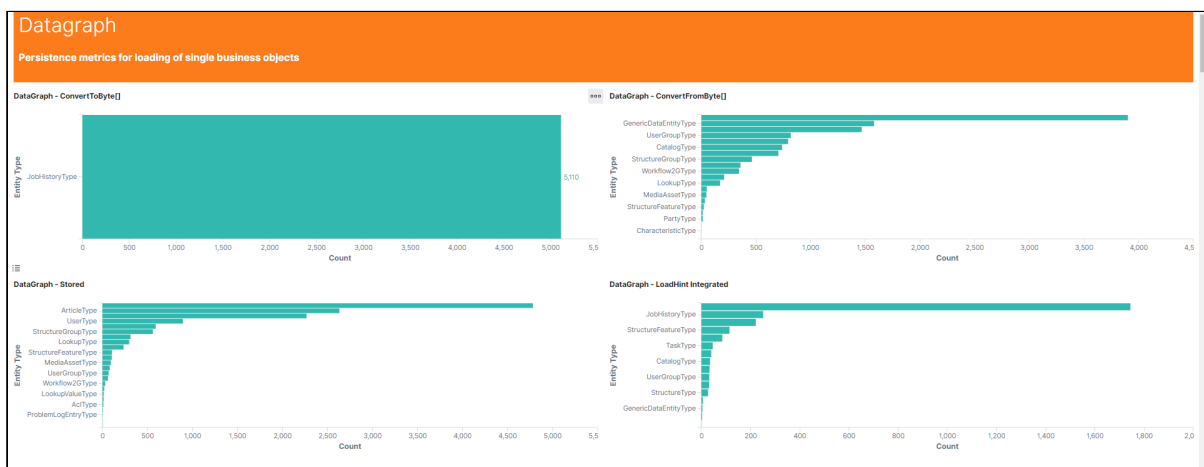
Δ EntityType
Select...

Δ Entity
Select...

Datagraph

Datagraph

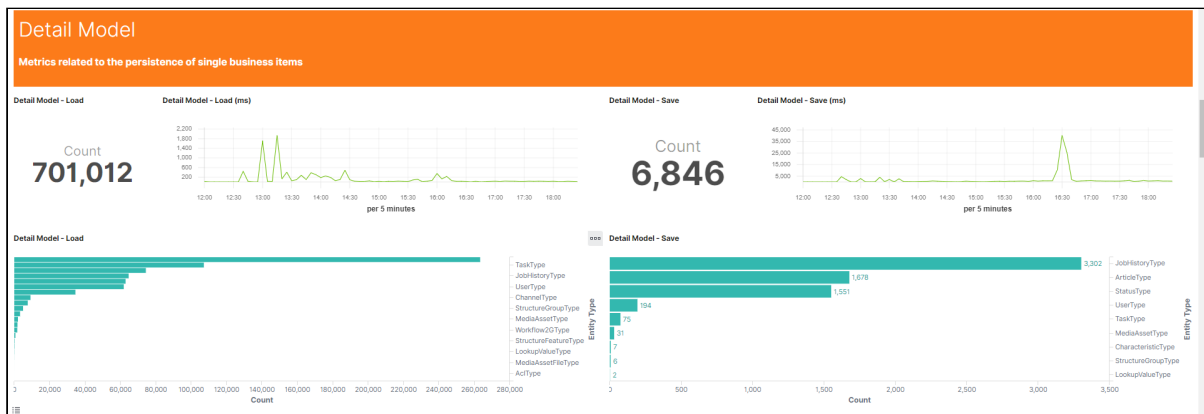
Persistence metrics for loading of single business objects.



Metric Name	Metric Key	Description
DataGraph - ConvertToByte[]	dataGraph_convertFromByteArray	The amount of datagraph models (i.e. a single business object) deserialised
DataGraph - ConvertFromByte[]	dataGraph_convertToByteArray	The amount of datagraph models (i.e. a single business object) serialised
DataGraph - LoadHin	dataGraph_loadHint_integrated	The amount of datagraph models (i.e. a single business object) read from the database
DataGraph - Stored	dataGraph_stored	The amount of datagraph models (i.e. a single business object) stored to the database

Detail Model Metrics

The Detail Model is the business object representation of a single object (e.g. a single item). A high load count is an indication that the application is loading a lot of single objects, while a high save count is an indication that the application is modifying a lot of single objects.

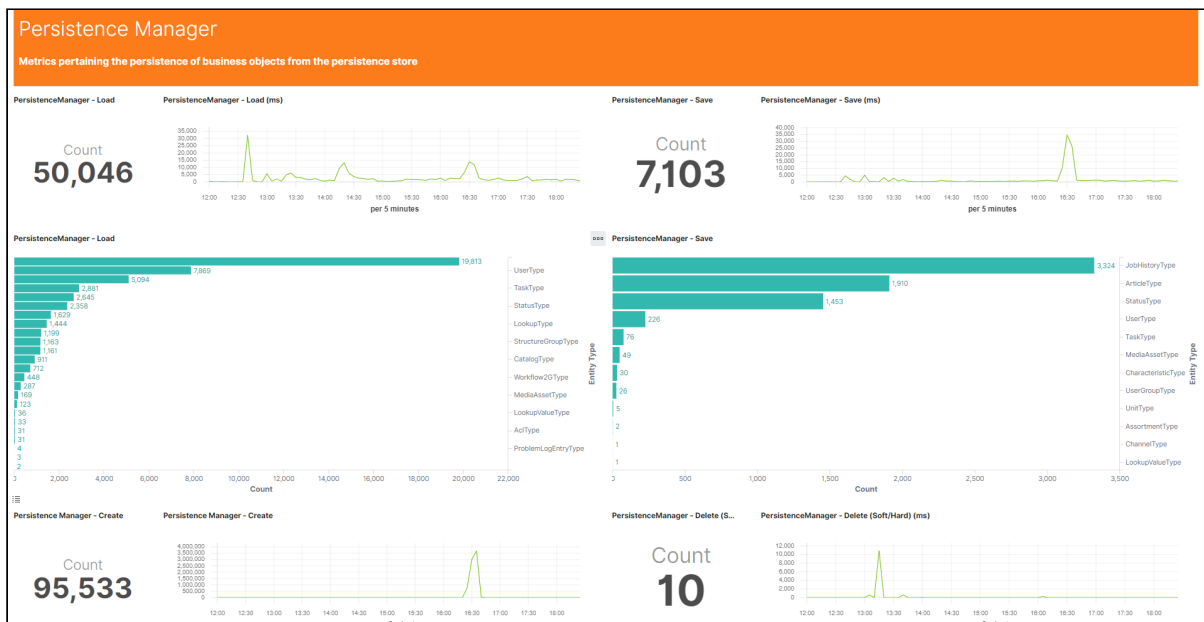


Metric Name	Metric Key	Description
Detail Model Load	detailModel_load	The amount of detail models (i.e. a full business entity) read from the database
Detail Model Save	detailModel_save	The amount of detail models (i.e. a full business entity) stored to the database

Persistence Manager

The PersistenceManager is the instance responsible for retrieving the business objects from the persistence store.

For example creating a new item will show up as **PersistenceManager - Create**, while loading an existing item will show up as **PersistenceManager - Load**.



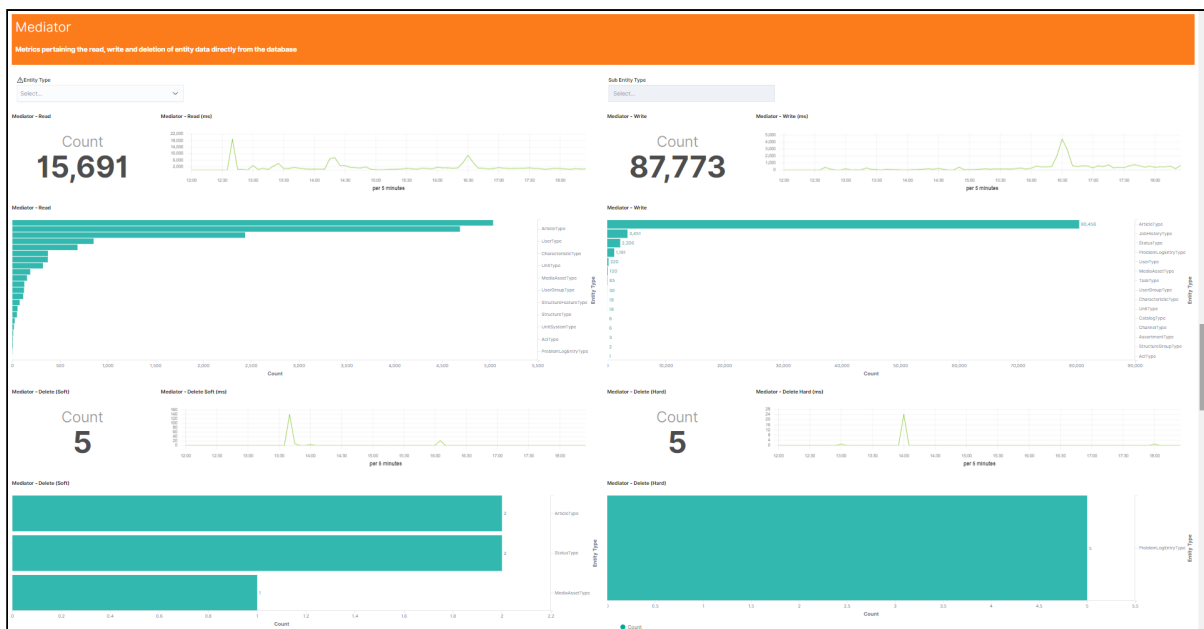
Metric Name	Metric Key	Description
PersistenceManager Create	persistenceManager_create	The amount of top level business entities created by the system
PersistenceManager Delete	persistenceManager_delete	The amount of top level business entities deleted by the system
PersistenceManager Read	persistenceManager_load	The amount of top level business entities loaded by the system
PersistenceManager Write	persistenceManager_save	The amount of top level business entities saved by the system

Mediator / Sub-Mediator

Mediators / Sub-Mediators are responsible for directly reading, writing and deleting Entity/ Sub-Entity data from the database.

For example creating a new item will show up as **Mediator/ SubMediator - Write**, while loading an existing item will show up as **Mediator/ SubMediator - Read**.

Sub-Mediator metrics are disabled by default to prevent excessive writes to the elastic search index. If required for debugging purposes, these metrics can be enabled by setting the corresponding log4j.xml entry to "INFO"



Metric Name	Metric Key	Description
Mediator Delete Hard	mediator_delete_hard	The amount of root entity hard delete operations
Mediator Delete Soft	mediator_delete_soft	The amount of root entity soft delete operations
Mediator Read	mediator_read	The amount of root entity load operations
Mediator Write	mediator_write	The amount of root entity save operations
SubMediator Delete Hard	submediator_delete_hard	The amount of sub entity hard (physical) delete operations
SubMediator Delete Soft	submediator_delete_soft	The amount of sub entity soft delete operations
SubMediator Read	submediator_read	The amount of sub entity load operations
SubMediator Write	submediator_write	The amount of sub entity save operations

8.2.2.5 Database Tables Statistics

The actual amount of rows of all configured database tables are available as metrics.

Since counting the rows of all tables in real time would be a very expensive operation, the row count is actually being read directly from the database statistics itself. The query which is being used to fetch the row count from the database statistics will be executed in a five minute interval, i.e. counters will be updated every five minutes. While Microsoft SQL Server usually has auto-update statistics, the statistics on Oracle databases usually must be refreshed periodically.

Database Tables Statistics				
The actual amount of rows of all configured database tables				
JobHistory 679	ProblemLogEntry 8,546	Workflow2G 14	ProcessStatusEntry 116	Report 413
Status 1,154,239	StatusEntry 19,874,377	Workflow2GStatus 30	Task 92	PublicationStatus 0

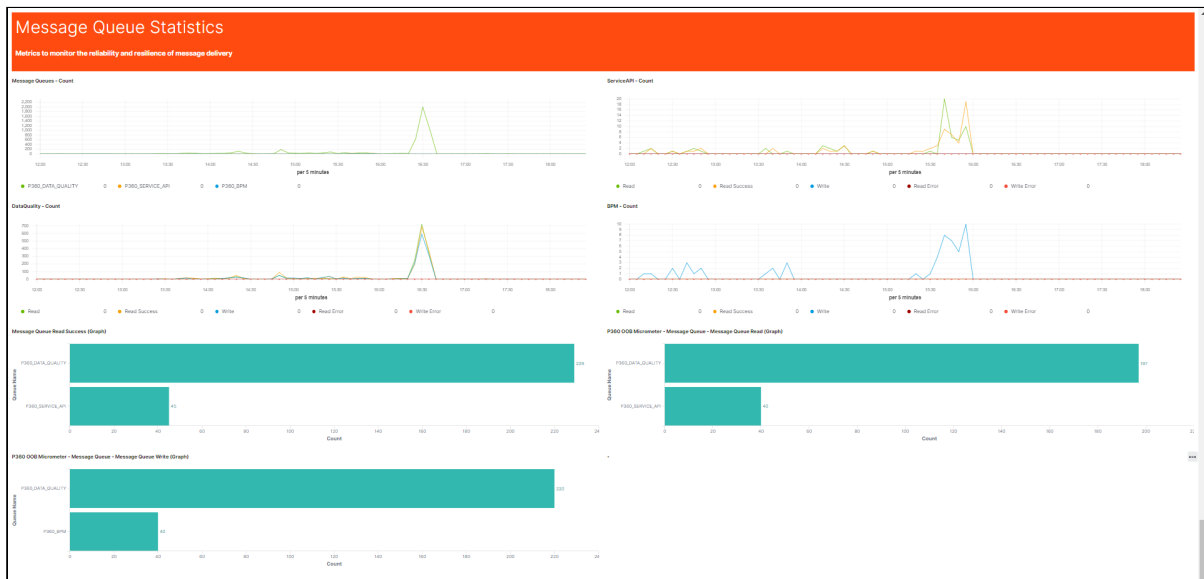
Metric Name	Metric Key	Description
Database Table Size	db_table_size	The amount of rows currently residing in the specified database table



Please note that for performance reasons this metric is being queried from the database statistics, so the real value could be different.

8.2.2.6 Message Queue Statistics

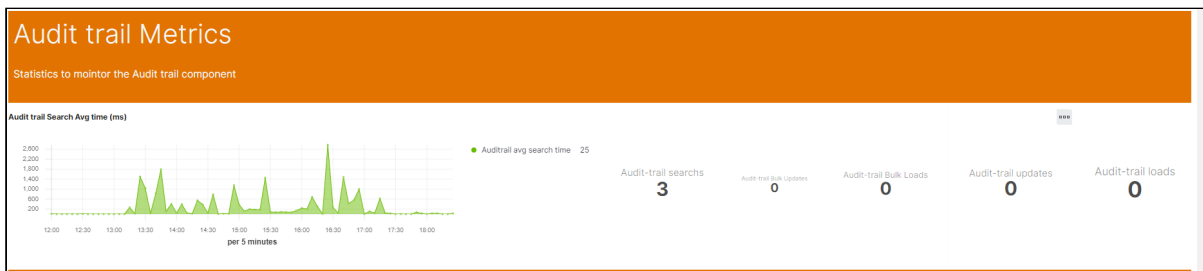
Metrics to monitor the reliability and resilience of message delivery in Service API, Data Quality and BPM message queues.



Metric Name	Metric Key	Description
Messages Read	queue_messages_read	The amount of messages read from the message queue
Messages Read Successfully	queue_messages_read_success	The amount of messages successfully read from the message queue
Messages Read Error	queue_messages_read_error	The amount of messages read with error from the message queue
Messages Write	queue_messages_write	The amount of messages written to the message queue
Messages Write Error	queue_messages_write_error	The amount of messages failed to write to the message queue

8.2.2.7 Audit trail Metrics

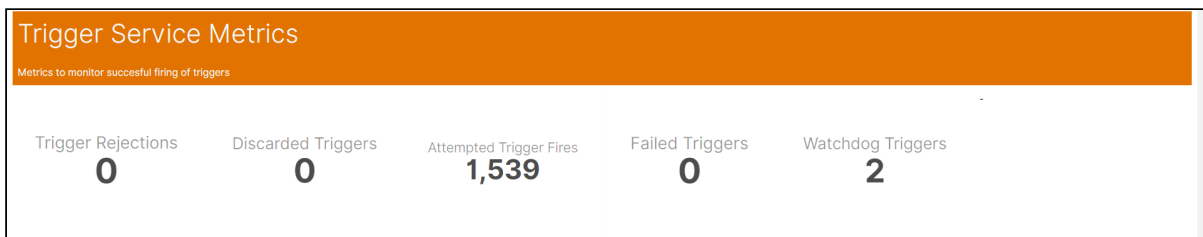
Statistics to monitor the Audit trail component.



Metric Name	Metric Key	Description
Audit trail search	auditTrail_searchService_search	timer measuring the average time taken for audit trail search
Audit trail create (Bulk)	auditTrailIndexService_bulkCreate	timer measuring bulk creations in the audit trail index
Audit trail update (Bulk)	auditTrailIndexService_bulkUpdate	timer measuring bulk updates to the audit trail index
Audit trail create	auditTrailIndexService_create	timer measuring individual creations in the audit trail index
Audit trail update	auditTrailIndexService_update	timer measuring updates to the audit trail index

8.2.2.8 Trigger Service Metrics

Metrics to monitor successful firing off triggers.



Metric Name	Metric Key	Description
Trigger Rejections	triggerService.rejectedAsExecutorServiceFull	It is the counter for number of requests that are directed to Trigger Rejection policy as Trigger executor service is under high load.
Discarded Triggers	triggerService.discardedAsNonJobThread	It is the counter for number of trigger execution

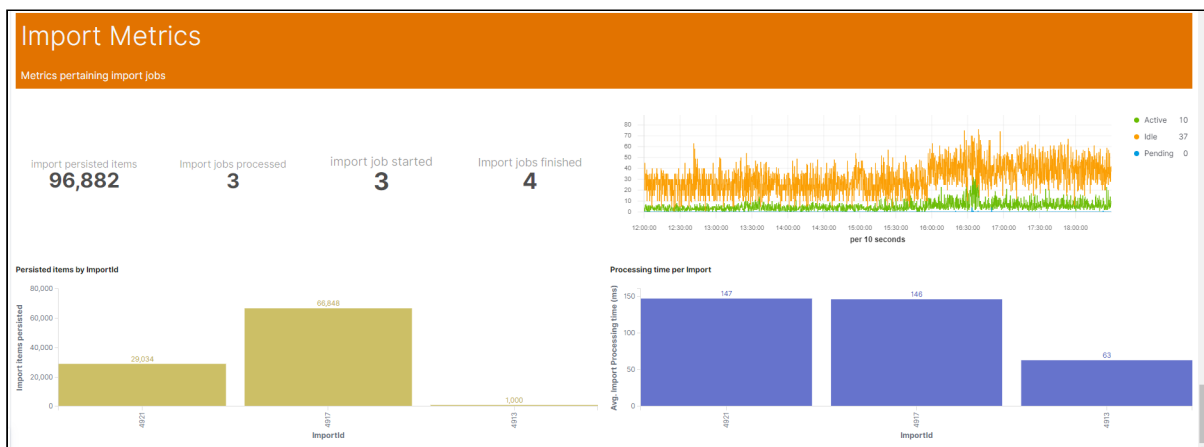
Metric Name	Metric Key	Description
		requests that are discarded as trigger service is under high load and this request is initiated from UI/OTHER initiator module. So these are the triggers that will be fired from trigger watchdog.
Watchdog Triggers	triggerService_pickedByTriggerWatchdog	Counter for number of triggers that are picked for firing by trigger watchdog
attemptedToFire	triggerService.attemptedToFire	Counter for number of triggers firing attempts made by Trigger Service
triggerFireFailed	triggerService.triggerFireFailed	Counter for number of trigger firing attempts that failed

8.2.2.9 Import Metrics

Metrics pertaining import jobs.



Please note import Statistics are only available from Product 360 10.1 HF1 onwards



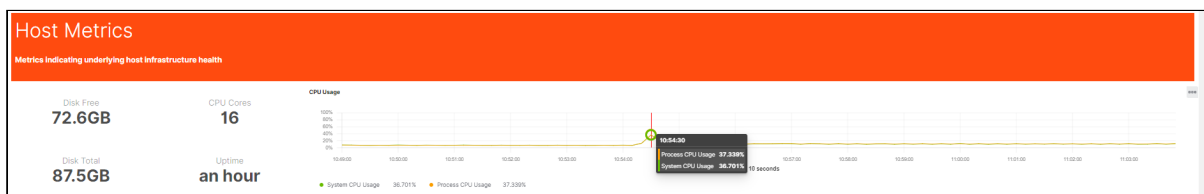
Metric Name	Metric Key	Description
Import - Items Persisted	importer_persistence_item	The amount of items persisted to the database during the import

Metric Name	Metric Key	Description
Import - Storage Get	importer_storage_get	The amount of objects retrieved from the import persistence model
Import - Storage Put	importer_storage_put	The amount of objects stored to the import persistence model
Import - Job Finished	importer_job_finished	The amount of finished import jobs
Import - Job Processing	importer_job_processed	The amount of currently active import jobs

8.2.3 System Metrics Dashboard

8.2.3.1 Host Metrics

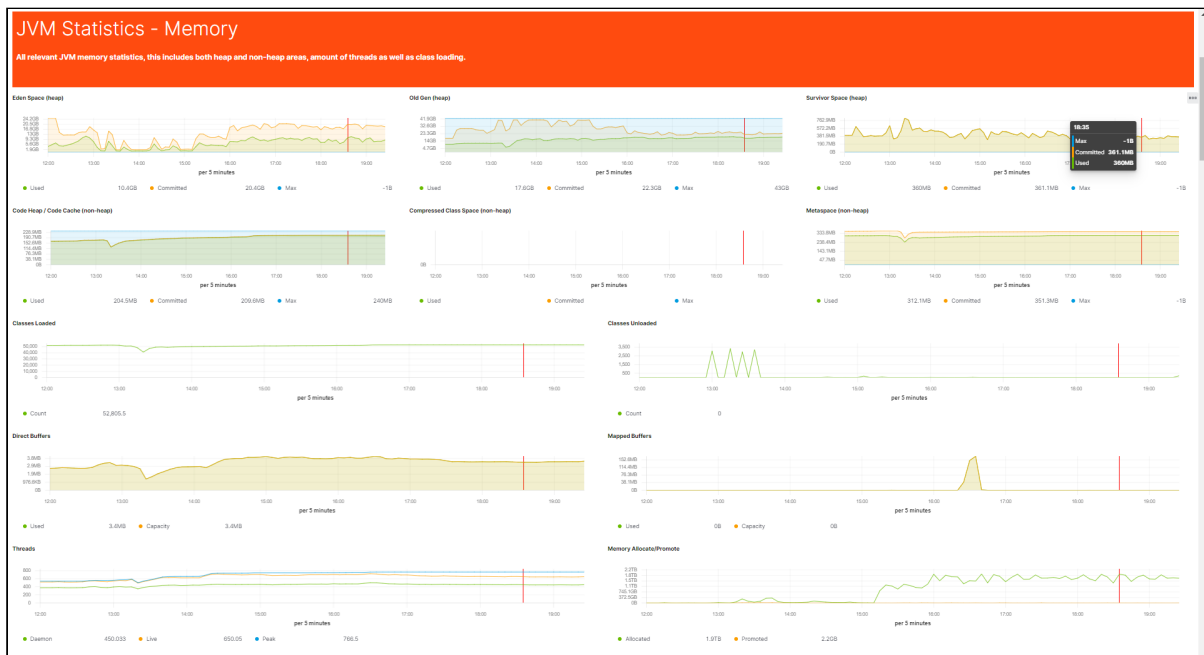
Metrics indicating underlying host infrastructure health.



Metric Name	Metric Key	Description
Disk Space Free	disk_free	The available disk space of the drive where Product 360 is installed
Disk Space Total	disk_total	The total disk space of the drive where Product 360 is installed
CPU Count	system_cpu_count	The available system processors
CPU Usage	system_cpu_usage	The current system CPU utilization
Process Uptime	process_uptime	The amount of time the Product 360 service is running

8.2.3.2 JVM Statistics - Memory

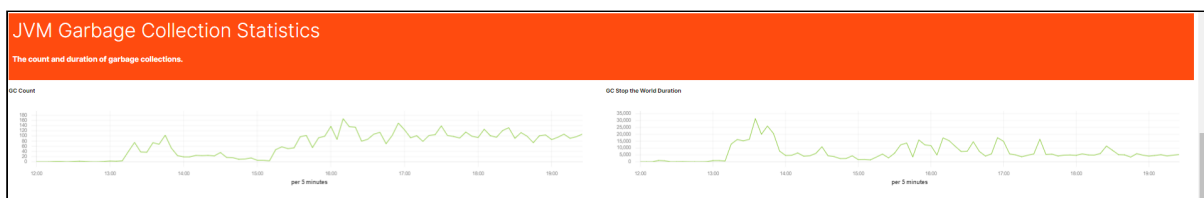
All relevant JVM memory statistics, this includes both heap and non-heap areas, amount of threads as well as class loading.



Metric Name	Metric Key	Description
Classes Loaded	jvm_classes_loaded	The amount of classes loaded by the JVM.
Classes Unloaded	jvm_classes_unloaded	The amount of classes unloaded by the JVM.
Buffer Memory	jvm_buffer_memory_used	The amount of buffer memory used by the JVM.
Memory Usage	jvm_memory_used	The amount of memory used by the JVM.
Threads	jvm_threads_daemon	The amount of threads used by the JVM.

8.2.3.3 JVM Garbage Collection Statistics

JVM garbage collection metrics include the count and duration of garbage collections. While a high count can be fine, a high duration usually suggests that the application is running low on heap memory.

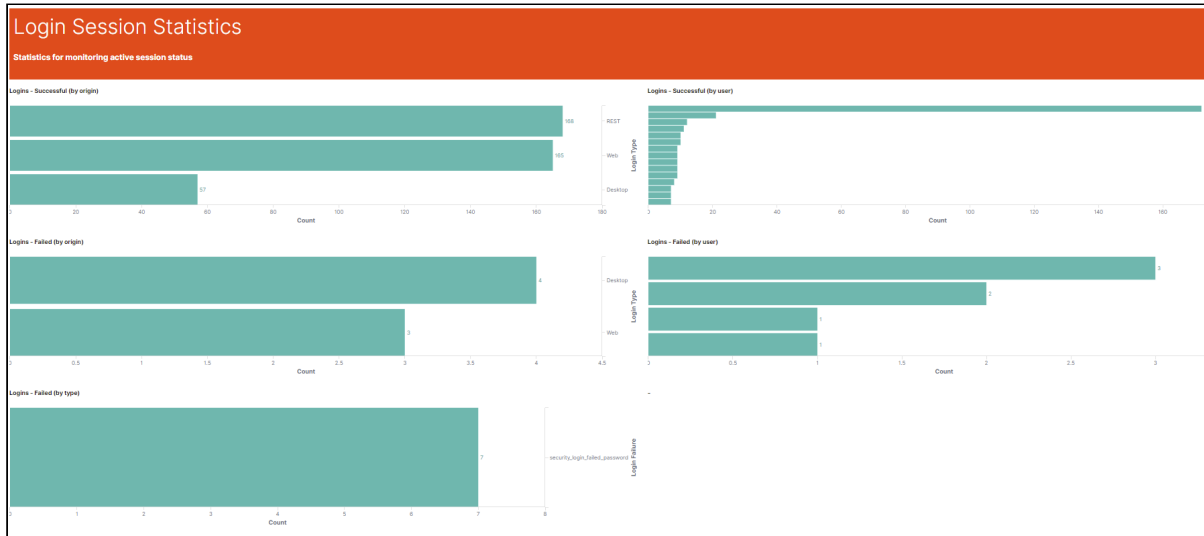


Metric Name	Metric Key	Description
GC Count	jvm_gc_pause	The amount of garbage collections executed by the JVM

Metric Name	Metric Key	Description
GC Stop the World Duration	jvm_gc_pause	The duration of the garbage collections executed by the JVM

8.2.3.4 Login Session Statistics

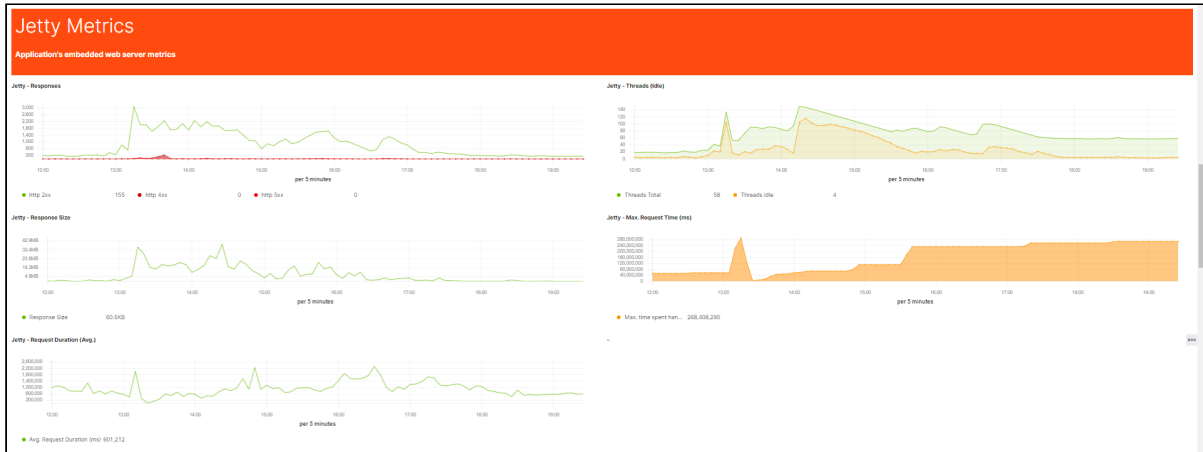
Statistics for monitoring active session status.



Metric Name	Metric Key	Description
Login Successful	security_login_successful	The amount of successful logins
Login Failed	security_login_failed_admin	The amount of failed logins for Administrator account
	security_login_failed_inactive	The amount of failed logins for inactive accounts
	security_login_failed_password	The amount of failed logins due to wrong password
	security_login_failed_permission	The amount of failed logins due to insufficient permissions
	security_login_failed_other	The amount of failed logins for any other reason

8.2.3.5 Jetty (Web Service Framework) Statistics

Application's embedded web server metrics.



Metric Name	Metric Key	Description
Request Duration	jetty_requests	The average duration of a web service request
Request Max. Time	jetty_dispatched_time_max	The maximum duration of a web service request
Response Size	jetty_responses_size	The size of response payload of the web service request
Responses	jetty_responses	The amount of responses to web service requests
Threads	jetty_threads_current	The current amount of JVM threads

8.2.3.6 Connection Pool Statistics

Metrics pertaining connection events such as creation, usage, return to connection pool etc.

Every data source has its own connection pool. A connection must be created, used, and returned to the connection pool and there are metrics for all events.

Important connection pool metrics are the amount of open connections, and how many of these connections are active or idling. It is important to watch for anything out of the ordinary, i.e. connection timeouts, many active connections and especially if any overflow of the connection pool takes place.

Global filter : Database

The database filter allows filtering (drill-down) of metrics regarding any kind of database activity. For example it is possible to show all active connections for a specific database.

i Please note that not all metrics are related to the database and won't show any data while the database has been selected.

Metric Tag

pool

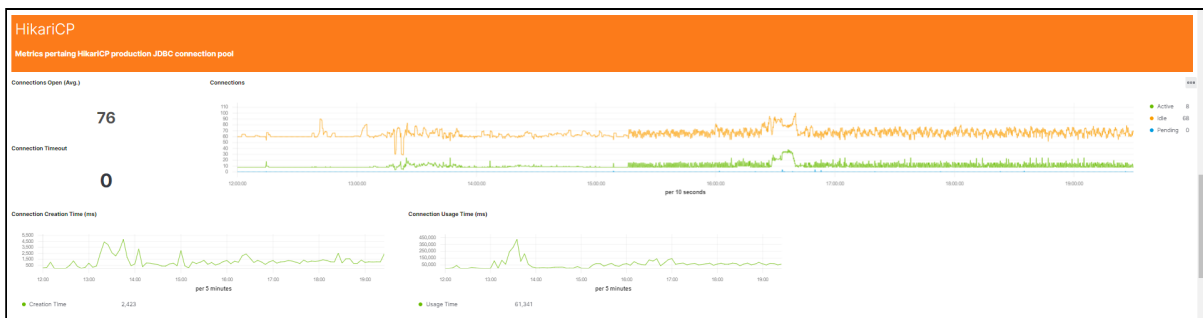
Connection Pool Statistics

Metrics pertaining connection events such as creation, usage, return to connection pool etc

Database
Select...

Database HikariCP Statistics

Metrics pertaining HikariCP production JDBC connection pool.

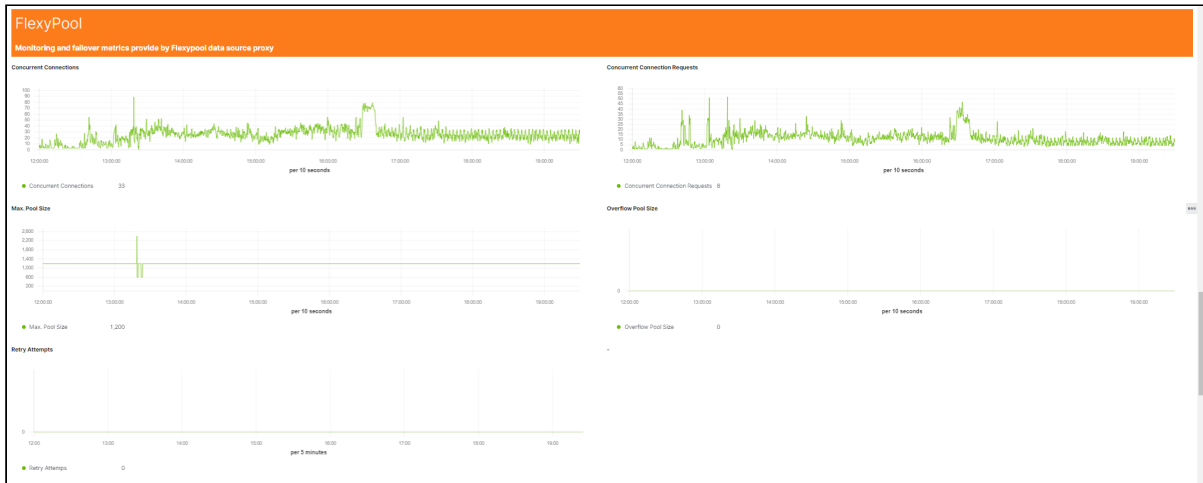


Metric Name	Metric Key	Description
Connection Acquire Time	hikaricp_connections_acquire	The amount of time it took to acquire a connection from the connection pool
Connection Creation Time	hikaricp_connections_creation	The amount of time it took to create a connection for the connection pool
Connection Timeout Count	hikaricp_connections_timeout	The amount of timeouts while requesting a connection from the connection pool
Connection Usage Time	hikaricp_connections_usage	The amount of time the connection from the connection pool was in use
Connections	hikaricp_connections_active	The amount of active connections in the connection pool

Metric Name	Metric Key	Description
Connections Size	hikaricp_connections	The amount of connections in the connection pool

Database FlexyPool Statistics

Monitoring and failover metrics provide by Flexypool data source proxy.



Metric Name	Metric Key	Description
Concurrent Connection Request	concurrentConnectionRequestsHistogram	The amount of concurrent connections requested by the connection pool
Concurrent Connections	concurrentConnectionsHistogram	The amount of concurrent connections used by the connection pool
Max Pool Size	maxPoolSizeHistogram	The maximum connection pool size
Overflow Pool Size	overflowPoolSizeHistogram	The overflow connection pool size (i.e. how much the maximum pool size has been exceeded)
Retry Attempts	retryAttemptsHistogram	The amount of retries required in order to request a connection from the connection pool

8.2.3.7 Thread Pool Statistics

Metrics for monitoring concurrent execution performance



Metric Name	Metric Key	Description
Executor Active	executor_active	The amount of active worker threads in the thread pool
Executor Completed	executor_completed	The amount of completed worker tasks of the thread pool
Executor Queued	executor_queued	The amount of queued worker tasks of the thread pool
Executor Pool Size	executor_pool_size	The thread pool size

Communication Framework


The Communication Framework is responsible for sending data between application servers and application server and rich clients.

Every communication message shows up in this thread pool, and it is important to watch out for queuing of any kind.

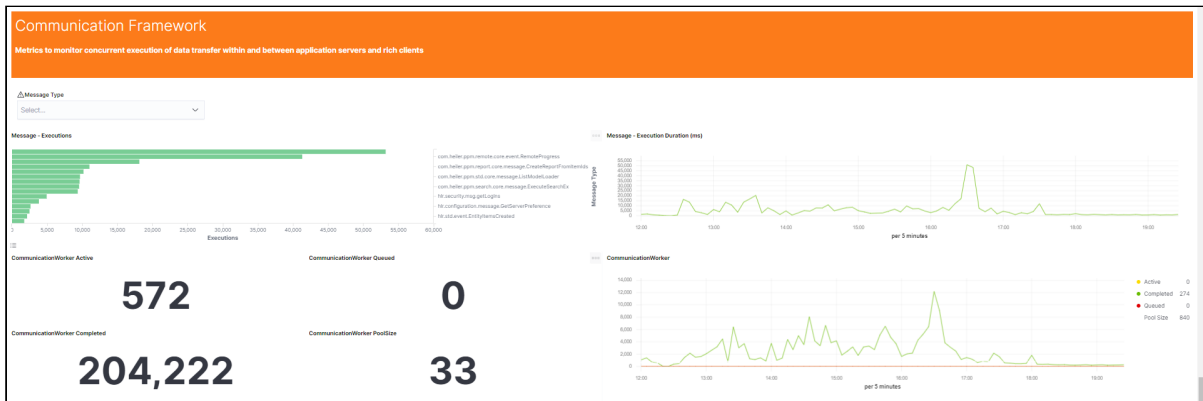
Global Filter : Message Type

The message type filters allows filtering (drill-down) of the communication framework for specific messages (every message in the communication framework has a specific message type).

For example it is possible to show the amount and duration of rich client which request data from the database.

 Please note that only the communication framework metrics are message type specific, all other metrics won't show any data while the message type has been selected.

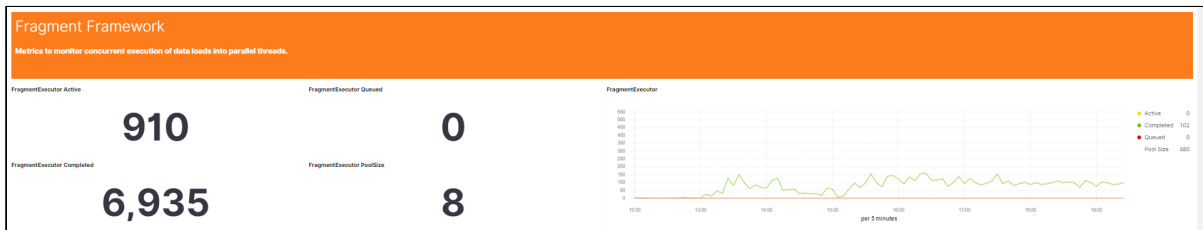
Metric Tag
messageType



Metric Name	Metric Key	Description
Message Execution	communication_message	The amount of messages executed by the system

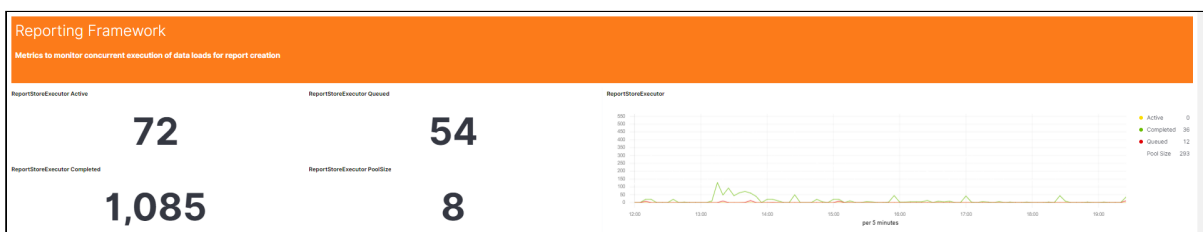
Fragment Framework

The Fragment Manager has a thread pool which supports loading data in parallel threads. Every parallel loading request of will show up in this thread pool, and it is important to watch out for huge queues.



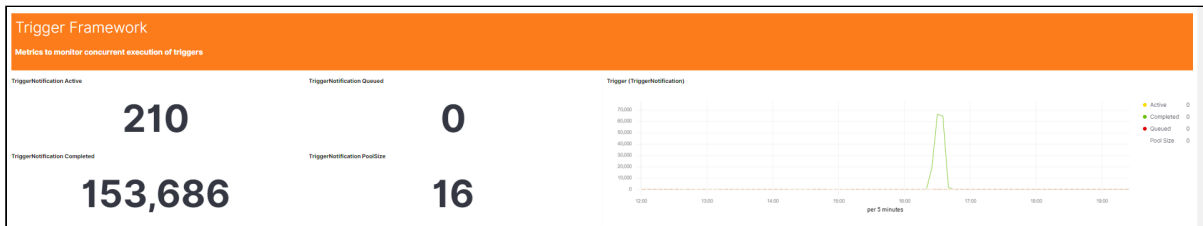
Reporting Framework

The Reporting Framework needs to create reports (i.e. list of object ids) in order to load any kind of data. When these reports exceed a specific size they will be created in parallel threads, and it is important to watch out for huge queues.



Trigger Framework

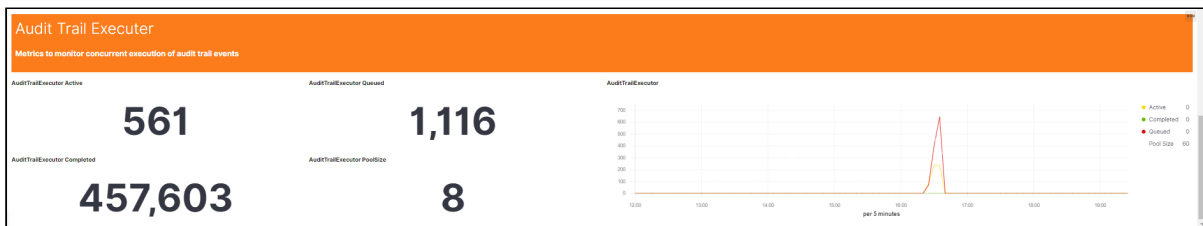
Metrics to monitor concurrent execution of triggers.



Audit Trail Executer

Every create, change or delete event in the application causes creation of an audit trail summary which will be consumed by triggers, workflows and audit trail.

Every audit trail event will show up in this thread pool, and it is important to watch out for huge queues.



8.2.4 System Metrics Dashboard - Extended

8.2.4.1 Cache Statistics

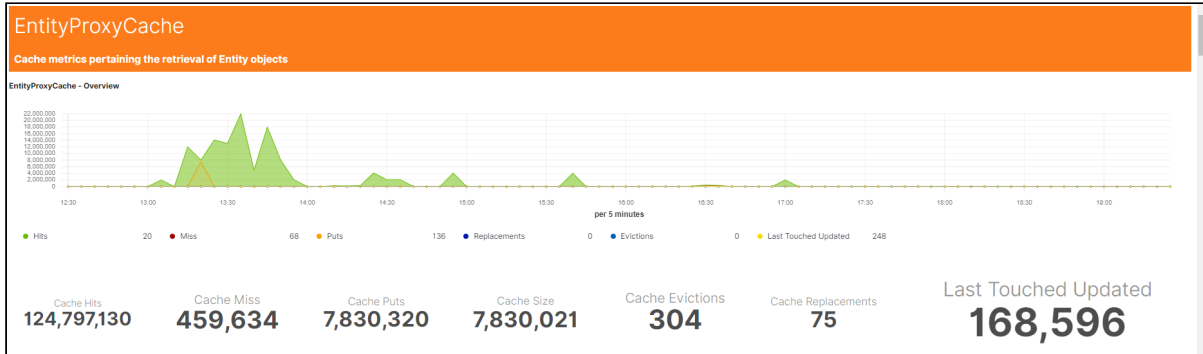
Metrics for various cache resources related to loading business objects. Cache objects expire upon modification or in case they have aged out. A high Cache miss rate should be investigated as well as Cache loads failure.



Metric Name	Metric Key	Description
Cache Evictions	cache_evictions	The amount of elements evicted by the cache
Cache Gets	cache_gets	The amount of elements requested by the cache
Cache Loads Duration	cache_load_duration	The amount of time it took the cache to load the element
Cache Loads	cache_load	The amount of elements loaded by the cache
Cache Puts	cache_puts	The amount of elements added to the cache
Cache Size	cache_size	The total amount of elements in the cache

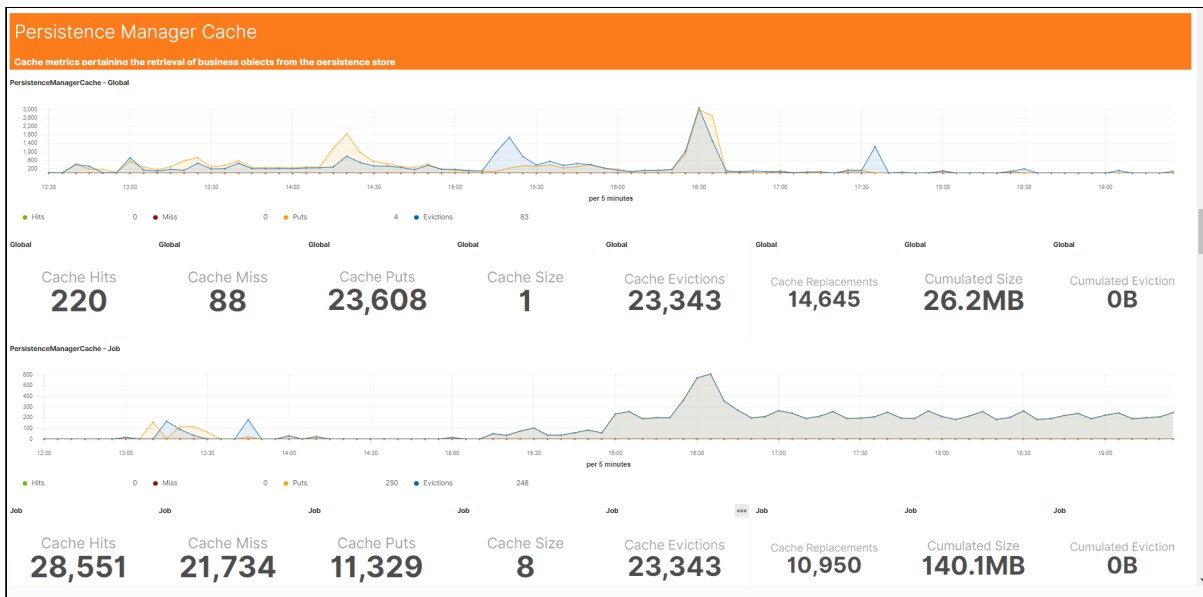
EntityProxyCache

Cache metrics pertaining the retrieval of Entity objects.



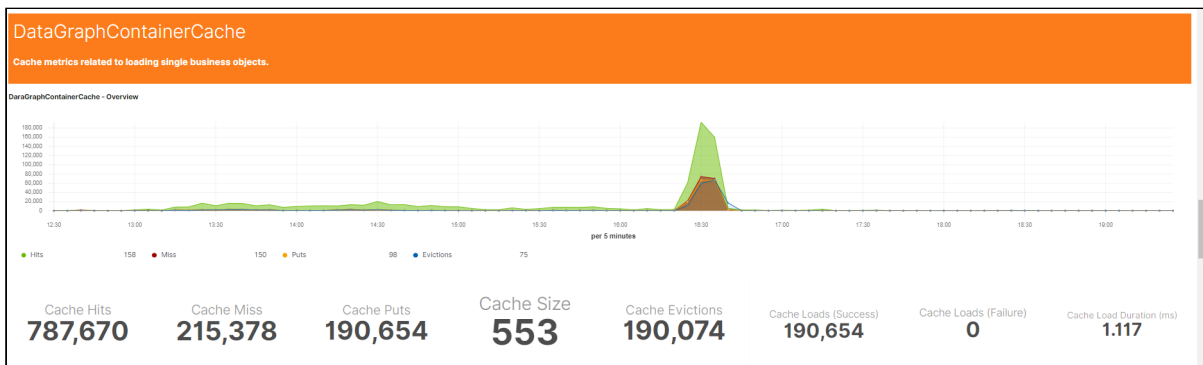
Persistence Manager Cache

Cache metrics pertaining the retrieval of business objects from the persistence store.



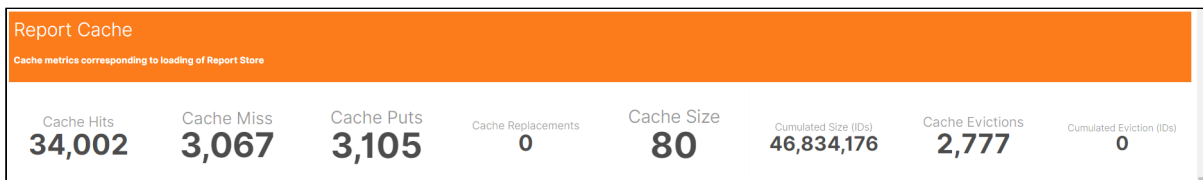
DataGraphContainerCache

Cache metrics related to loading single business objects.



Report Cache

Cache metrics corresponding to loading of Report Store.



8.2.4.2 Hibernate Statistics

Metrics for all database change events triggered by queries, statements, connections, sessions, transactions, flushes etc.

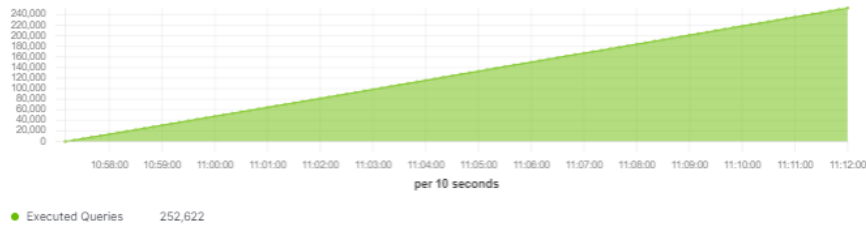
Hibernate is the persistence service of the application and responsible for loading, saving and deleting data from the databases. Any change to the database requires a query or a statement, a connection, a session, a transaction, a flush, and there are metrics for all events.

Important Hibernate metrics are the amount of queries executed and especially queries which take a lot of time.

Hibernate Statistics

Metrics for all database change events triggered by queries, statements, connections, sessions, transactions, flushes etc

Slowest Query (ms)
14648.00



Executed Queries
252,622

Open Sessions
505,696

Transactions (Success)
211,058

Statements Prepared
3,939,880

Flushes
421,936

Obtained Connections
505,693

Transactions (Failure)
0

Statements Closed
0

Entities Loads
5,285,351

Entities Inserts
218,296

Entities Updates
1,836,880

Entities Deletes
0

Collections Loads
3,948,283

Collections Inserts
0

Collections Updates
126,414

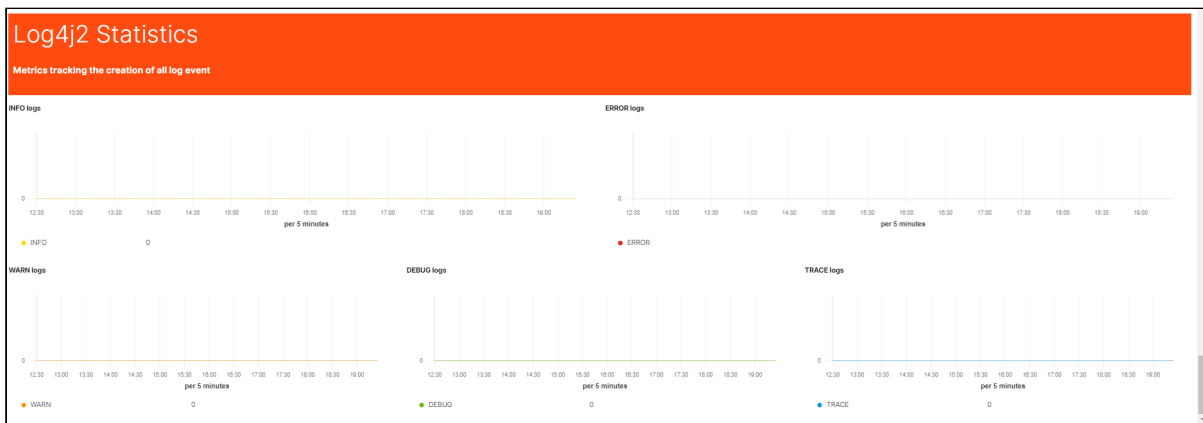
Collections Deletes
0

Metric Name	Metric Key	Description
Collections Deletes	hibernate_collections_deletes	The amount of collections (i.e. records from 1:n relationships) deleted by Hibernate
Collections Inserts	hibernate_collections_inserts	The amount of collections (i.e. records from 1:n relationships) created by Hibernate
Collections Loads	hibernate_collections_loads	The amount of collections (i.e. records from 1:n relationships) loaded by Hibernate
Collections Updates	hibernate_collections_updates	The amount of collections (i.e. records from 1:n relationships) updated by Hibernate
Connections Obtained	hibernate_connections_obtained	The amount of database connections obtained by Hibernate
Entities Deletes	hibernate_entities_deletes	The amount of entities (i.e. top level business entities) deleted by Hibernate

Metric Name	Metric Key	Description
Entitites Inserts	hibernate_entities_inserts	The amount of collections (i.e. top level business entities) created by Hibernate
Entitites Loads	hibernate_entities_loads	The amount of collections (i.e. top level business entities) loaded by Hibernate
Entitites Updates	hibernate_entities_updates	The amount of collections (i.e. top level business entities) updated by Hibernate
Flushes	hibernate_flushes	The amount of flushes (i.e. synchronizing the database with the current state) executed by Hibernate
Query Executions	hibernate_query_executions	The amount of single database queries executed by Hibernate <i>Please note that enabling Hibernate Query Metrics results in noticable performance penalties</i>
Query Max. Execution Time	hibernate_query_executions_max	The maximum duration of each single database query executed by Hibernate <i>Please note that enabling Hibernate Query Metrics results in noticable performance penalties</i>
Query Max. Rows	hibernate_query_executions_rows	The maximum amount of rows retrieved of each single database query executed by Hibernate <i>Please note that enabling Hibernate Query Metrics results in noticable performance penalties</i>
Session Open	hibernate_sessions_open	The amount of sessions opened (i.e. acquiring a connection to the database) by Hibernate
Statements Closed/ Prepared	hibernate_statements	The amount of statements prepared and closed by Hibernate
Transactions	hibernate_transactions	The amount of transactions executed by Hibernate

8.2.4.3 Log4j2 Logging Statistics

Metrics tracking the creation of all log event. The creation of any log event (TRACE to ERROR) will be shown [here](#).



Metric Name	Metric Key	Description
DEBUG Logs	log4j2_events (level:debug)	The amount of DEBUG log entries written
ERRORS Logs	log4j2_events (level:error)	The amount of ERROR log entries written
INFO Logs	log4j2_events (level:info)	The amount of INFO log entries written
TRACE Logs	log4j2_events (level:trace)	The amount of TRACE log entries written
WARN Logs	log4j2_events (level:warn)	The amount of WARNING log entries written

8.3 Best Practices and Recommendations

For the optimal usage of Micrometer metrics and the Kibana dashboard, the following recommendations are made:

- Ensure that **elastic.step** in **micrometer.properties** server settings is not less than 1 min, as the defaults mentioned earlier. This setting controls the frequency with which metrics are stored to the elastic search index. Too short an interval could lead to increased elasticsearch index sizes
- The "**Informatica MDM - Product 360 System Metrics (Extended)**" gives access to low level monitoring of Caches, Hibernate etc. These metrics are disabled by default to prevent excessive elasticsearch index sizes and performance issues. These metrics should be enabled primarily for debugging purposes.
- When querying statistics for a time range of over multiple days, ensure that the "Maximum buckets" setting in **Kibana > Management > Advanced Settings** is set to a high number (greater than 20000 as suggested in the defaults)

Copyright

Copyright (c) 1993-2020 Informatica LLC. All rights reserved.

This Software and documentation contain proprietary information of Informatica LLC and are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright law. Reverse engineering of the software is prohibited. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC. This Software may be protected by [U.S. and/or](#) international Patents and other Patents Pending.

Use, duplication, or disclosure of the Software by the U.S. Government is subject to the restrictions set forth in the applicable software license agreement and as provided in DFARS 227.7202-1(a) and 227.7702-3(a) (1995), DFARS 252.227-7013©(1)(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14

(ALT III), as applicable.

The information in this product or documentation is subject to change without notice. If you find any problems in this product or documentation, please report them to us in writing.

Informatica, Informatica Platform, Informatica Data Services, PowerCenter, PowerCenterRT, PowerCenter Connect, PowerCenter Data Analyzer, PowerExchange, PowerMart, Metadata Manager, Informatica Data Quality, Informatica Data Explorer, Informatica B2B Data Transformation, Informatica B2B Data Exchange Informatica On Demand, Informatica Identity Resolution, Informatica Application Information Lifecycle Management, Informatica Complex Event Processing, Ultra Messaging and Informatica Master Data Management are trademarks or registered trademarks of Informatica LLC in the United States and in jurisdictions throughout the world. All other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties, including without limitation: Copyright DataDirect Technologies. All rights reserved. Copyright © Sun Microsystems. All rights reserved. Copyright © RSA Security Inc. All Rights Reserved. Copyright © Ordinal Technology Corp. All rights reserved. Copyright © Aandacht c.v. All rights reserved. Copyright Genivia, Inc. All rights reserved. Copyright Isomorphic Software. All rights reserved. Copyright © Meta Integration Technology, Inc. All rights reserved. Copyright © Intalio. All rights reserved. Copyright © Oracle. All rights reserved. Copyright © Adobe Systems Incorporated. All rights reserved. Copyright © DataArt, Inc. All rights reserved. Copyright © ComponentSource. All rights reserved. Copyright © Microsoft Corporation. All rights reserved. Copyright © Rogue Wave Software, Inc. All rights reserved. Copyright © Teradata Corporation. All rights reserved. Copyright © Yahoo! Inc. All rights reserved. Copyright © Glyph & Cog, LLC. All rights reserved. Copyright © Thinkmap, Inc. All rights reserved. Copyright © Clearpace Software Limited. All rights reserved. Copyright © Information Builders, Inc. All rights reserved. Copyright © OSS Nokalva, Inc. All rights reserved. Copyright Edifecs, Inc. All rights reserved.

Copyright Cleo Communications, Inc. All rights reserved. Copyright © International Organization for Standardization 1986. All rights reserved. Copyright © ej-technologies GmbH. All rights reserved. Copyright © Jaspersoft Corporation. All rights reserved. Copyright © International Business Machines Corporation. All rights reserved. Copyright © yWorks GmbH. All rights reserved. Copyright © Lucent Technologies. All rights reserved. Copyright (c) University of Toronto. All rights reserved. Copyright © Daniel Veillard. All rights reserved. Copyright © Unicode, Inc. Copyright IBM Corp. All rights reserved. Copyright © MicroQuill Software Publishing, Inc. All rights reserved. Copyright © PassMark Software Pty Ltd. All rights reserved. Copyright © LogiXML, Inc. All rights reserved. Copyright © 2003-2010 Lorenzi Davide, All rights reserved. Copyright © Red Hat, Inc. All rights reserved. Copyright © The Board of Trustees of the Leland Stanford Junior University. All rights reserved. Copyright

© EMC Corporation. All rights reserved. Copyright © Flexera Software. All rights reserved. Copyright © Jinfonet Software. All rights reserved. Copyright © Apple Inc. All rights reserved. Copyright © Telerik Inc. All rights reserved. Copyright © BEA Systems. All rights reserved. Copyright © PDFlib GmbH. All rights reserved. Copyright © Orientation in Objects GmbH. All rights reserved. Copyright © Tanuki Software, Ltd. All rights reserved. Copyright © Ricebridge. All rights reserved. Copyright © Sencha, Inc. All rights reserved. Copyright © Scalable Systems, Inc. All rights reserved. Copyright © jqWidgets. All rights reserved. Copyright © Tableau Software, Inc. All rights reserved. Copyright © MaxMind, Inc. All Rights Reserved. Copyright © TMate Software s.r.o. All rights reserved. Copyright © MapR Technologies Inc. All rights reserved.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>), and/or other software which is licensed under various versions of the Apache License (the "License"). You may obtain a copy of these Licenses at <http://www.apache.org/licenses/>. Unless required by applicable law or agreed to in writing, software distributed under these Licenses is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the Licenses for the specific language governing permissions and limitations under the Licenses.

This product includes software which was developed by Mozilla (<http://www.mozilla.org/>), software copyright The JBoss Group, LLC, all rights reserved; software copyright © 1999-2006 by Bruno Lowagie and Paulo Soares and other software which is licensed under various versions of the GNU Lesser General Public License Agreement, which may be found at [http:// www.gnu.org/licenses/lgpl.html](http://www.gnu.org/licenses/lgpl.html). The materials are provided free of charge by Informatica, "as-is", without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

The product includes ACE(TM) and TAO(TM) software copyrighted by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University, Copyright (©) 1993-2006, all rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (copyright The OpenSSL Project. All Rights Reserved) and redistribution of this software is subject to terms available at <http://www.openssl.org> and <http://www.openssl.org/source/license.html>.

This product includes Curl software which is Copyright 1996-2013, Daniel Stenberg, < daniel@haxx.se >. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at [http:// curl.haxx.se/docs/ copyright.html](http://curl.haxx.se/docs/copyright.html). Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

The product includes software copyright 2001-2005 (©) MetaStuff, Ltd. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at [http://www.dom4j.org/ license.html](http://www.dom4j.org/license.html).

The product includes software copyright © 2004-2007, The Dojo Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at [http://dojotoolkit.org/ license](http://dojotoolkit.org/license).

This product includes ICU software which is copyright International Business Machines Corporation and others. All rights reserved. Permissions and limitations regarding this software are subject to terms available at <http://source.icu-project.org/repos/icu/icu/trunk/license.html>.

This product includes software copyright © 1996-2006 Per Bothner. All rights reserved. Your right to use such materials is set forth in the license which may be found at [http://www.gnu.org/software/kawa/ Software-License.html](http://www.gnu.org/software/kawa/Software-License.html).

This product includes OSSP UUID software which is Copyright © 2002 Ralf S. Engelschall, Copyright © 2002 The OSSP Project Copyright © 2002 Cable & Wireless Deutschland. Permissions and limitations regarding this software are subject to terms available at <http://www.opensource.org/licenses/mit-license.php>.

This product includes software developed by Boost (<http://www.boost.org/>) or under the Boost software license. Permissions and limitations regarding this software are subject to terms available at [http:// www.boost.org/LICENSE_1_0.txt](http://www.boost.org/LICENSE_1_0.txt).

This product includes software copyright © 1997-2007 University of Cambridge. Permissions and limitations regarding this software are subject to terms available at <http://www.pcre.org/license.txt>. This product includes software copyright © 2007 The Eclipse Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at [http:// www.eclipse.org/org/documents/ epl-v10.php](http://www.eclipse.org/org/documents/epl-v10.php) and at <http://www.eclipse.org/org/documents/edl-v10.php>.

This product includes software licensed under the terms at <http://www.tcl.tk/software/tcltk/license.html>, <http://www.bosrup.com/web/overlib/?License>, [http:// www.stlport.org/doc/ license.html](http://www.stlport.org/doc/license.html), <http://asm.ow2.org/license.html>, <http://www.cryptix.org/LICENSE.TXT>, <http://hsqldb.org/web/hsqLicense.html>, <http://unit.sourceforge.net/doc/license.html>, <http://jung.sourceforge.net/license.txt>, http://www.gzip.org/zlib/zlib_license.html, [http://www.openldap.org/software/release/ license.html](http://www.openldap.org/software/release/license.html), <http://www.libssh2.org>, <http://slf4j.org/license.html>, <http://www.sente.ch/software/OpenSourceLicense.html>, [http:// fusesource.com/downloads/license- agreements/fuse-message-broker-v-5-3- licenseagreement](http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3-licenseagreement); [http:// antlr.org/license.html](http://antlr.org/license.html); <http://aopalliance.sourceforge.net/>; <http://www.bouncycastle.org/licence.html>; <http://www.jgraph.com/jgraphdownload.html>; <http://www.jcraft.com/jsch/LICENSE.txt>; http://jotm.objectweb.org/bsd_license.html; [http://www.w3.org/Consortium/Legal/ 2002/copyright-software-20021231](http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231);

www.slf4j.org/license.html; <http://nanoxml.sourceforge.net/orig/copyright.html>; <http://www.json.org/license.html>; <http://forge.ow2.org/projects/jaservice/>, <http://www.postgresql.org/about/licence.html>, <http://www.sqlite.org/copyright.html>, <http://www.tcl.tk/software/tcltk/license.html>, <http://www.jaxen.org/faq.html>, <http://www.jdom.org/docs/faq.html>, <http://www.slf4j.org/license.html>; <http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/License>; <http://www.keplerproject.org/md5/license.html>; <http://www.toedter.com/en/jcalendar/license.html>; <http://www.edankert.com/bounce/index.html>; <http://www.net-snmp.org/about/license.html>; <http://www.openmdx.org/#FAQ>; http://www.php.net/license/3_01.txt; <http://srp.stanford.edu/license.txt>; <http://www.schneier.com/blowfish.html>; <http://www.jmock.org/license.html>; <http://xsom.java.net>; <http://benalman.com/about/license/>; <https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js>; <http://www.h2database.com/html/license.html#summary>; <http://jsoncpp.sourceforge.net/LICENSE>; <http://jdbc.postgresql.org/license.html>; <http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto>; <https://github.com/rantav/hector/blob/master/LICENSE>; <http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html>; <http://jibx.sourceforge.net/jibx-license.html>; <https://github.com/lyokato/libgeohash/blob/master/LICENSE>; <https://github.com/hjiang/jsonxx/blob/master/LICENSE>; <https://code.google.com/p/lz4/>; <https://github.com/jedisct1/libsodium/blob/master/LICENSE>; <http://one-jar.sourceforge.net/index.php?page=documents&file=license>; <https://github.com/EsotericSoftware/kryo/blob/master/license.txt>; <http://www.scala-lang.org/license.html>; <https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt>; and <http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html>.

This product includes software licensed under the Academic Free License (<http://www.opensource.org/licenses/afl-3.0.php>), the Common Development and Distribution License (<http://www.opensource.org/licenses/cddl1.php>) the Common Public License (<http://www.opensource.org/licenses/cpl1.0.php>), the Sun Binary Code License Agreement Supplemental License Terms, the BSD License (<http://www.opensource.org/licenses/bsd-license.php>), the new BSDLicense (<http://opensource.org/licenses/BSD-3-Clause>), the MIT License (<http://www.opensource.org/licenses/mitlicense.php>), the Artistic License (<http://www.opensource.org/licenses/artistic-license-1.0>) and the Initial Developer's Public License Version 1.0 (<http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/>).

This product includes software copyright © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. All rights reserved. Permissions and limitations regarding this software are subject to terms available at <http://xstream.codehaus.org/license.html>. This product includes software developed by the Indiana University Extreme! Lab. For further information please visit <http://www.extreme.indiana.edu/>.

This product includes software Copyright (c) 2013 Frank Balluffi and Markus Moeller. All rights reserved. Permissions and limitations regarding this software are subject to terms of the MIT license.

This Software is protected by U.S. Patent Numbers 5,794,246; 6,014,670; 6,016,501; 6,029,178; 6,032,158; 6,035,307; 6,044,374; 6,092,086; 6,208,990; 6,339,775; 6,640,226; 6,789,096; 6,823,373; 6,850,947; 6,895,471; 7,117,215; 7,162,643; 7,243,110; 7,254,590; 7,281,001; 7,421,458; 7,496,588; 7,523,121; 7,584,422; 7,676,516; 7,720,842; 7,721,270; 7,774,791; 8,065,266; 8,150,803; 8,166,048; 8,166,071; 8,200,622; 8,224,873; 8,271,477; 8,327,419; 8,386,435; 8,392,460; 8,453,159; 8,458,230; 8,707,336; 8,886,617 and RE44,478, International Patents and other Patents Pending.

DISCLAIMER: Informatica LLC provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.iv>

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.

2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY,

MISREPRESENTATION AND OTHER TORTS.