



Informatica®  
10.1.1

# SQL Data Service Guide

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. 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.

Informatica, the Informatica logo, and PowerCenter are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>. 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 © 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. Copyright © Amazon Corporate LLC. All rights reserved. Copyright © Highsoft. All rights reserved. Copyright © Python Software Foundation. All rights reserved. Copyright © BeOpen.com. All rights reserved. Copyright © CNRI. 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>. 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](mailto: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>. 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>.

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>.

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> 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://asm.ow2.org/license.html>, <http://www.cryptix.org/LICENSE.TXT>, <http://hsqldb.org/web/hsqldbLicense.html>, <http://httpunit.sourceforge.net/doc/license.html>, <http://jung.sourceforge.net/license.txt>, [http://www.gzip.org/zlib/zlib\\_license.html](http://www.gzip.org/zlib/zlib_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-license-agreement>, <http://antlr.org/license.html>, <http://aopalliance.sourceforge.net/>, <http://www.bouncycastle.org/license.html>, <http://www.jgraph.com/jgraphdownload.html>, <http://www.jcraft.com/jsch/LICENSE.txt>, [http://jotm.objectweb.org/bsd\\_license.html](http://jotm.objectweb.org/bsd_license.html), <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>, <http://www.slf4j.org/license.html>, <http://nanoxml.sourceforge.net/org/copyright.html>, <http://www.json.org/license.html>, <http://forge.ow2.org/projects/javaservice/>, <http://www.postgresql.org/about/license.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://www.php.net/license/3_01.txt), <http://www.srp.stanford.edu/license.txt>, <http://www.schneider.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/tinkerpops/blueprints/blob/master/LICENSE.txt>; <http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html>; <https://aws.amazon.com/asl/>; <https://github.com/twbs/bootstrap/blob/master/LICENSE>; <https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt>; <https://github.com/documentcloud/underscore-contrib/blob/master/LICENSE>, and <https://github.com/apache/hbase/blob/master/LICENSE.txt>.

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 BSD License (<http://opensource.org/licenses/BSD-3-Clause>), the MIT License (<http://www.opensource.org/licenses/mit-license.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.

See patents at <https://www.informatica.com/legal/patents.html>.

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.

## 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.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, please report them to us in writing at Informatica LLC 2100 Seaport Blvd. Redwood City, CA 94063.

INFORMATICA LLC PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2018-06-09

# Table of Contents

<b>Preface .....</b>	<b>8</b>
Informatica Resources. ....	8
Informatica Network. ....	8
Informatica Knowledge Base. ....	8
Informatica Documentation. ....	8
Informatica Product Availability Matrixes. ....	9
Informatica Velocity. ....	9
Informatica Marketplace. ....	9
Informatica Global Customer Support. ....	9
 <b>Chapter 1: Introduction to SQL Data Services.....</b>	<b>10</b>
SQL Data Services Overview. ....	10
SQL Data Service Connections. ....	10
SQL Data Service Example. ....	11
 <b>Chapter 2: Virtual Data.....</b>	<b>12</b>
Virtual Data Overview. ....	12
SQL Data Services. ....	13
Defining an SQL Data Service. ....	13
Creating an SQL Data Service. ....	13
Virtual Tables. ....	14
Data Access Methods. ....	15
Creating a Virtual Table from a Data Object. ....	15
Creating a Virtual Table Manually. ....	16
Defining Relationships between Virtual Tables. ....	17
Running an SQL Query to Preview Data. ....	17
Persisting Virtual Data in Temporary Tables. ....	17
Temporary Table Implementation. ....	18
Temporary Table Operations. ....	18
Rules and Guidelines for Temporary Tables. ....	20
Virtual Table Mappings. ....	20
Defining a Virtual Table Mapping. ....	21
Creating a Virtual Table Mapping. ....	21
Validating a Virtual Table Mapping. ....	21
Previewing Virtual Table Mapping Output. ....	22
Virtual Stored Procedures. ....	22
Defining a Virtual Stored Procedure. ....	23
Creating a Virtual Stored Procedure. ....	23
Validating a Virtual Stored Procedure. ....	24
Previewing Virtual Stored Procedure Output. ....	24

SQL Query Plans. . . . .	24
SQL Query Plan Optimization. . . . .	24
SQL Query Plan Optimization Examples. . . . .	25
Viewing an SQL Query Plan. . . . .	26

## **Chapter 3: SQL Syntax..... 27**

SQL Syntax Overview. . . . .	27
Datatypes. . . . .	28
Operators. . . . .	28
Functions. . . . .	28
SQL Statements and Keywords. . . . .	36
Queries. . . . .	37
Non-Correlated Subqueries. . . . .	37
Correlated Subqueries. . . . .	38
Parameterized Queries. . . . .	39
Reserved Words. . . . .	39
Escape Syntax. . . . .	39
Escape Syntax for the fn Keyword. . . . .	40
Troubleshooting SQL Data Services. . . . .	42

## **Chapter 4: Installing and Configuring Drivers for Third-Party Clients..... 44**

Installing and Configuring Drivers for Third-Party Clients Overview. . . . .	44
Before You Install the Drivers. . . . .	45
Extract the Installer Files. . . . .	45
Record Informatica Domain Information. . . . .	45
Configure the Client for Secure Communication. . . . .	46
Configure the Client for Kerberos Authentication. . . . .	47
Install the Drivers on Windows. . . . .	47
Install the Drivers in Graphical Mode. . . . .	47
Install the Drivers in Silent Mode. . . . .	48
Install the Drivers on UNIX. . . . .	48
Install the Drivers in Console Mode. . . . .	48
Install the Drivers in Silent Mode. . . . .	49
JDBC Connections. . . . .	50
JDBC Connection Properties. . . . .	50
Configure JDBC Connections on Windows. . . . .	51
Configure JDBC Connections on UNIX. . . . .	52
Troubleshooting JDBC Connections. . . . .	52
ODBC Connections. . . . .	53
ODBC Connection Properties. . . . .	53
Configure ODBC Connections on Windows. . . . .	55
Configure ODBC Connections on UNIX. . . . .	56

<b>Chapter 5: Third-Party Client Tool Configuration.....</b>	<b>58</b>
Third-Party Client Tool Configuration Overview. . . . .	58
BusinessObjects. . . . .	59
BusinessObjects Configuration. . . . .	59
IBM Cognos Configuration. . . . .	60
Step 1. Update the IBM Cognos Configuration File. . . . .	61
Step 2. Create the ODBC Connection and Import Metadata. . . . .	61
MicroStrategy Configuration. . . . .	65
Step 1. Create the Database Instance and Connection. . . . .	65
Step 2. Configure the SQL Generation Options. . . . .	70
Oracle Business Intelligence Enterprise Edition 11g Configuration. . . . .	71
Oracle Database Gateway Configuration. . . . .	72
QlikView Configuration. . . . .	72
SQL Server Business Intelligence Development Studio Configuration. . . . .	74
Squirrel SQL Client Configuration. . . . .	75
Step 1. Copy the Driver File. . . . .	75
Step 2. Create the Driver and Alias. . . . .	76
Tableau Configuration. . . . .	78
Toad for Data Analysts Configuration. . . . .	78
WinSQL Configuration. . . . .	79
Troubleshooting Third-Party Client Tools. . . . .	80
 <b>Chapter 6: Installing and Configuring Drivers for PowerCenter.....</b>	 <b>82</b>
Installing and Configuring Drivers for PowerCenter Overview. . . . .	82
Before You Install the Drivers. . . . .	82
Extract the Installer Files. . . . .	83
Record Informatica Domain Information. . . . .	83
Install the Drivers on Windows. . . . .	84
Install the Drivers for PowerCenter in Graphical Mode. . . . .	84
Install the Drivers for PowerCenter in Silent Mode. . . . .	85
Configure ODBC Connections on Windows. . . . .	85
Install the Drivers on UNIX. . . . .	86
Install the Drivers for PowerCenter in Console Mode. . . . .	87
Install the Drivers for PowerCenter in Silent Mode. . . . .	87
Configure ODBC Connections on UNIX. . . . .	88
Configure the Shared Library Environment Variable. . . . .	88
Configure the ODBC Files. . . . .	89
Troubleshooting ODBC Connections for PowerCenter. . . . .	89
 <b>Chapter 7: SQL Data Service Administration.....</b>	 <b>91</b>
SQL Data Service Administration Overview. . . . .	91
SQL Data Service Security Management. . . . .	92

Types of SQL Data Service Permissions. . . . .	92
Column Level Security. . . . .	93
Row Level Security. . . . .	93
Pass-Through Security. . . . .	93
SQL Data Service Properties Configuration . . . . .	94
SQL Data Service Properties. . . . .	94
SQL Properties. . . . .	96
Virtual Table Properties. . . . .	97
Virtual Column Properties. . . . .	98
Virtual Stored Procedure Properties. . . . .	98
Logical Data Objects. . . . .	98
Logical Data Object/Virtual Table Cache PropertiesLogical Data Object Cache Properties. . . . .	99
SQL Data Service Result Set Caching. . . . .	100
Data Object Caching. . . . .	101
SQL Data Service Logs. . . . .	102
Monitor SQL Data Services. . . . .	102
Properties View for an SQL Data Service. . . . .	102
Connections View for an SQL Data Service. . . . .	103
Requests View for an SQL Data Service. . . . .	103
Virtual Tables View for an SQL Data Service. . . . .	104
Reports View for an SQL Data Service. . . . .	105
<b>Index. . . . .</b>	<b>106</b>

# Preface

The *Informatica SQL Data Service Guide* discusses SQL data services, virtual data, configuration, connecting to an SQL data service with a third-party tool, and troubleshooting. It also provides instructions on these concepts. This guide is intended for data service developers. It assumes that you have an understanding of flat files and relational databases in your environment.

## Informatica Resources

### Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit <https://network.informatica.com>.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

### Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit <https://kb.informatica.com>. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at [KB\\_Feedback@informatica.com](mailto:KB_Feedback@informatica.com).

### Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at [https://kb.informatica.com/\\_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx](https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx).

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at [infa\\_documentation@informatica.com](mailto:infa_documentation@informatica.com).



## Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at

<https://network.informatica.com/community/informatica-network/product-availability-matrices>.

## Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at <http://velocity.informatica.com>.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at [ips@informatica.com](mailto:ips@informatica.com).

## Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at <https://marketplace.informatica.com>.

## Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:

<http://www.informatica.com/us/services-and-training/support-services/global-support-centers>.

If you are an Informatica Network member, you can use Online Support at <http://network.informatica.com>.

# CHAPTER 1

## Introduction to SQL Data Services

This chapter includes the following topics:

- [SQL Data Services Overview, 10](#)
- [SQL Data Service Connections, 10](#)
- [SQL Data Service Example, 11](#)

### SQL Data Services Overview

A data service is a collection of reusable operations that you can run to access and transform data. An SQL data service is a virtual database that end users can query. It contains virtual schemas and the virtual tables or stored procedures that define the database structure.

Create an SQL data service so that end users can run SQL queries against the virtual tables through a third-party client tool. End users can query the virtual tables as if they were physical tables. End users can also use a third-party client tool to run virtual stored procedures.

You can create an SQL data service in the Developer tool. To make it available to end users, include it in an application, and deploy the application to a Data Integration Service. When the application is running, end users can connect to the SQL data service from a third-party client tool by supplying a connect string. After they connect to the SQL data service, end users can run SQL queries through the client tool.

### SQL Data Service Connections

An SQL data service is a virtual database that you can query. It provides a uniform view of data that might be scattered among disparate, heterogeneous data sources. You can use a third-party JDBC or ODBC client tool to run SQL queries against the virtual tables in an SQL data service and to run virtual stored procedures.

An SQL data service can contain the following objects:

- Virtual schemas that define the database structure
- Virtual tables
- Virtual table mappings that define the flow of data between sources and a virtual table
- Virtual stored procedures that take optional inputs through parameters, transform the data, and optionally return output through parameters

To make the virtual tables and virtual stored procedures available to you, a developer creates and deploys an application that contains the SQL data service. The developer deploys the application to a Data Integration Service and an administrator runs the application. When the application is running, you can query the virtual tables and run the virtual stored procedures in the SQL data service.

When you query a virtual table or run a virtual stored procedure, the JDBC or ODBC driver sends the request to the Data Integration Service. By default, the driver uses HTTP to communicate with the Data Integration Service. If an administrator enables Transport Layer Security (TLS) for the domain, the driver uses TLS to communicate with the Data Integration Service.

Before you can run SQL queries or virtual stored procedures for the first time, you must configure the machine from which you want to access the SQL data service. You must also configure the client tool so that it can connect to the SQL data service.

## SQL Data Service Example

Create an SQL data service to make a virtual database available for end users to query. Create a virtual database to define uniform views of data and to isolate the data from changes in structure. For example, create an SQL data service to define a uniform view of customer data and to allow end users to run SQL queries against the data.

Two companies that store customer data in multiple, heterogeneous data sources merge. A developer at the merged company needs to make a single view of customer data available to other users at the company. The other users need to make SQL queries against the data to retrieve information such as the number of customers in a region or a list of customers whose purchases exceed a certain dollar amount.

To accomplish this goal, the developer creates an SQL data service that contains virtual schemas and virtual tables that define a unified view of a customer. The developer creates virtual table mappings to link the virtual tables of the customer with the sources and to standardize the data. To make the virtual data accessible by end users, the developer includes the SQL data service in an application and deploys the application.

After the developer deploys the application, end users can make SQL queries against the standardized view of the customer through a JDBC or ODBC client tool.

## CHAPTER 2

# Virtual Data

This chapter includes the following topics:

- [Virtual Data Overview, 12](#)
- [SQL Data Services, 13](#)
- [Virtual Tables, 14](#)
- [Persisting Virtual Data in Temporary Tables, 17](#)
- [Virtual Table Mappings, 20](#)
- [Virtual Stored Procedures, 22](#)
- [SQL Query Plans, 24](#)

## Virtual Data Overview

Create a virtual database to define uniform views of data and make the data available for end users to query. End users can run SQL queries against the virtual tables as if they were physical database tables.

Create a virtual database to accomplish the following tasks:

- Define a uniform view of data that you can expose to end users.
- Define the virtual flow of data between the sources and the virtual tables. Transform and standardize the data.
- Provide end users with access to the data. End users can use a JDBC or ODBC client tool to run SQL queries against the virtual tables as if they were actual, physical database tables.
- Isolate the data from changes in data structures. You can add the virtual database to a self-contained application. If you make changes to the virtual database in the Developer tool, the virtual database in the application does not change until you redeploy it.

To create a virtual database, you must create an SQL data service. An SQL data service contains the virtual schemas and the virtual tables or stored procedures that define the database structure. If the virtual schema contains virtual tables, the SQL data service also contains virtual table mappings that define the flow of data between the sources and the virtual tables.

After you create an SQL data service, you add it to an application and deploy the application to make the SQL data service accessible by end users.

End users can query the virtual tables or run the stored procedures in the SQL data service by entering an SQL query in a third-party client tool. When the user enters the query, the Data Integration Service retrieves virtual data from the sources or from cache tables, if an administrator specifies that any of the virtual tables should be cached.

**Note:** A virtual table mapping will fail if it contains a user-defined parameter.

## SQL Data Services

An SQL data service is a virtual database that end users can query. It contains a schema and other objects that represent underlying physical data.

An SQL data service can contain the following objects:

- Virtual schemas. Schemas that define the virtual database structure.
- Virtual tables. The virtual tables in the database. You can create virtual tables from physical or logical data objects, or you can create virtual tables manually.
- Virtual table mappings. Mappings that link a virtual table to source data and define the data flow between the sources and the virtual table. If you create a virtual table from a data object, you can create a virtual table mapping to define data flow rules between the data object and the virtual table. If you create a virtual table manually, you must create a virtual table mapping to link the virtual table with source data and define data flow.
- Virtual stored procedures. Sets of data flow instructions that allow end users to perform calculations or retrieve data.

### Defining an SQL Data Service

To define an SQL data service, create an SQL data service and add objects to it.

1. Create an SQL data service.  
You can create virtual tables and virtual table mappings during this step.
2. Create virtual tables in the SQL data service.  
You can create a virtual table from a data object, or you can create a virtual table manually.
3. Define relationships between virtual tables.
4. Create or update virtual table mappings to define the data flow between data objects and the virtual tables.
5. Optionally, create virtual stored procedures.
6. Optionally, preview virtual table data.

### Creating an SQL Data Service

Create an SQL data service to define a virtual database that end users can query. When you create an SQL data service, you can create virtual schemas, virtual tables, and virtual table mappings that link virtual tables with source data.

1. Select a project or folder in the **Object Explorer** view.
2. Click **File > New > Data Service**.  
The **New** dialog box appears.
3. Select **SQL Data Service**.
4. Click **Next**.

5. Enter a name for the SQL data service.
6. To create virtual tables in the SQL data service, click **Next**. To create an SQL data service without virtual tables, click **Finish**.  
If you click Next, the **New SQL Data Service** dialog box appears.
7. To create a virtual table, click the New button.  
The Developer tool adds a virtual table to the list of virtual tables.
8. Enter a virtual table name in the Name column.
9. Click the Open button in the Data Object column.  
The **Select a Data Object** dialog box appears.
10. Select a physical or logical data object and click **OK**.
11. Enter the virtual schema name in the Virtual Schema column.
12. Select **Read** in the Data Access column to link the virtual table with the data object. Select **None** if you do not want to link the virtual table with the data object.
13. Repeat steps [7](#) through [12](#) to add more virtual tables.
14. Click **Finish**.  
The Developer tool creates the SQL data service.

## Virtual Tables

A virtual table is a table in a virtual database. Create a virtual table to define the structure of the data.

Create one or more virtual tables within a schema. If a schema contains multiple virtual tables, you can define primary key-foreign key relationships between tables.

You can create virtual tables manually or from physical or logical data objects. Each virtual table has a data access method. The data access method defines how the Data Integration Service retrieves data. When you manually create a virtual table, the Developer tool creates an empty virtual table and sets the data access method to none.

When you create a virtual table from a data object, the Developer tool creates a virtual table with the same columns and properties as the data object. The Developer tool sets the data access method to read. If you change columns in the data object, the Developer tool updates the virtual table with the same changes. The Developer tool does not update the virtual table if you change the data object name or description.

To define data transformation rules for the virtual table, set the data access method to custom. The Developer tool prompts you to create a virtual table mapping.

You can preview virtual table data when the data access method is read or custom.

## Data Access Methods

The data access method for a virtual table defines how the Data Integration Service retrieves data.

When you create a virtual table, you must choose a data access method. The following table describes the data access methods:

Data Access Method	Description
None	<p>The virtual table is not linked to source data.</p> <p>If you change the data access method to none, the Developer tool removes the link between the data object and the virtual table. If the virtual table has a virtual table mapping, the Developer tool deletes the virtual table mapping.</p> <p>The Data Integration Service cannot retrieve data for the table.</p>
Read	<p>The virtual table is linked to a physical or logical data object without data transformation. If you add, remove, or change a column in the data object, the Developer tool makes the same change to the virtual table. However, if you change primary key-foreign key relationships, change the name of the data object, or change the data object description, the Developer tool does not update the virtual table.</p> <p>If you change the data access method to read, the Developer tool prompts you to choose a data object. If the virtual table has a virtual table mapping, the Developer tool deletes the virtual table mapping.</p> <p>When an end user queries the virtual table, the Data Integration Service retrieves data from the data object.</p>
Custom	<p>The virtual table is linked to a physical or logical data object through a virtual table mapping. If you update the data object, the Developer tool does not update the virtual table.</p> <p>If you change the data access method to custom, the Developer tool prompts you to create a virtual table mapping.</p> <p>When an end user queries the virtual table, the Data Integration Service applies any transformation rule defined in the virtual table mapping to the source data. It returns the transformed data to the end user.</p>

## Creating a Virtual Table from a Data Object

Create a virtual table from a physical or logical data object when the virtual table structure matches the structure of the data object. The Developer tool creates a virtual table mapping to read data from the data object.

1. Open an SQL data service.
2. Click the **Schema** view.
3. Drag a physical or logical data object from the **Object Explorer** view to the editor.

The **Add Data Objects to SQL Data Service** dialog box appears. The Developer tool lists the data object in the Data Object column.

4. Enter the virtual schema name in the Virtual Schema column.
5. Click **Finish**.

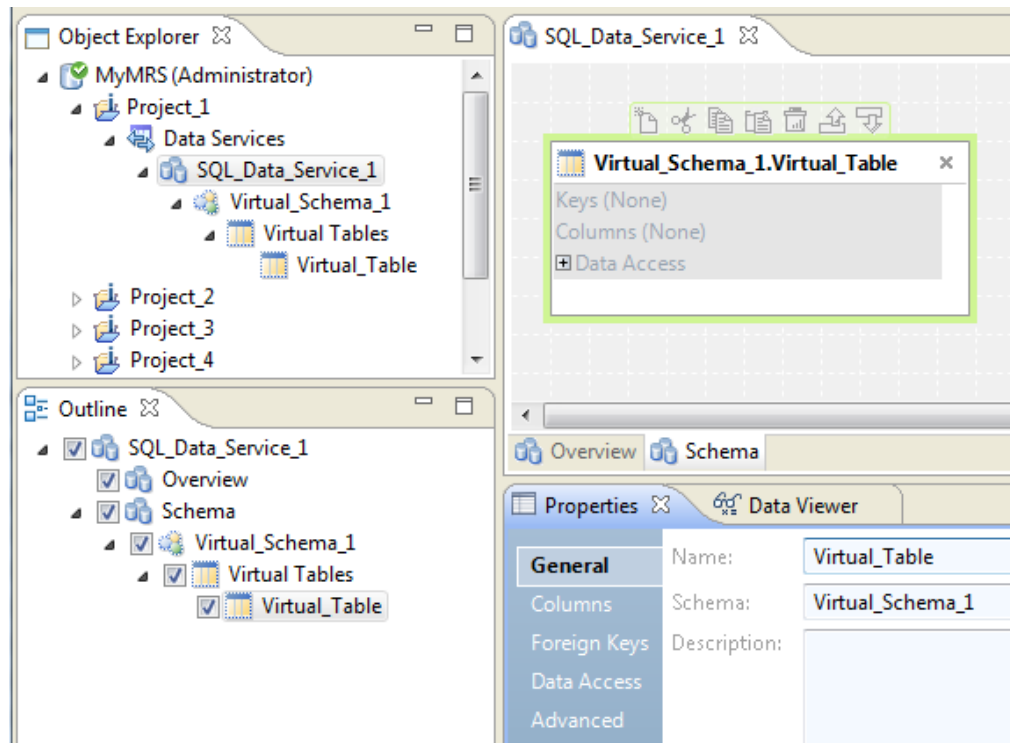
The Developer tool places the virtual table in the editor and sets the data access method to read.

## Creating a Virtual Table Manually

Create a virtual table manually when the virtual table structure does not match the structure of an existing data object. The Developer tool sets the data access method for the virtual table to none, which indicates the virtual table is not linked to a source.

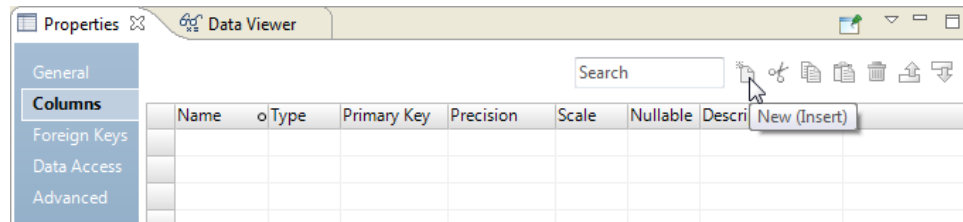
1. Open an SQL data service.
2. In the **Overview** view **Tables** section, click the New button.  
The **New Virtual Table** dialog box appears.
3. Enter a name for the virtual table.
4. Enter a virtual schema name or select a virtual schema.
5. Click **Finish**.

The following image shows the virtual table in the **Schema** view:



6. To add a column to the virtual table, perform the following steps:
  - a. Select **Data Access** in the **Properties** view. Verify that the **Access Method** is set to Custom.
  - b. Select **Columns** in the **Properties** view and click **New**.

The following image shows the location of the **New** button.



Verify that the virtual column names are not reserved words for the SQL standard.

7. To make a column a primary key, click the blank space to the left of the column name.



## Defining Relationships between Virtual Tables

You can define primary key-foreign key relationships between virtual tables in an SQL data service to show associations between columns in the virtual tables.

1. Open an SQL data service.
2. Expand the SQL data service. Select one of the virtual tables.
3. In the **Outline** view, select **Foreign Key**.  
The Virtual Table editor displays the **Foreign Keys** view, with a list of available foreign keys.
4. In the **Foreign Keys** view, click **Add**.  
The **Create New Foreign Key** dialog box opens.
5. Select the table that has the primary key that you want to use.
6. In the Foreign Key column, select the field you want to use as a foreign key.
7. Click **Finish**.

## Running an SQL Query to Preview Data

Run an SQL query against a virtual table to preview the data.

For the query to return results, the virtual table must be linked to source data. Therefore, the virtual table must be created from a data object or it must be linked to source data in a virtual table mapping.

1. Open an SQL data service.
2. Click the **Schema** view.
3. Select the virtual table in the **Outline** view.  
The virtual table appears in the **Schema** view.
4. Click the **Data Viewer** view.
5. Enter an SQL statement in the **Input** window.  
For example:  

```
select * from <schema>.<table>
```
6. Click **Run**.  
The query results appear in the **Output** window.

## Persisting Virtual Data in Temporary Tables

A temporary table is a table in a relational database that stores intermediate, temporary data. Complex queries commonly require storage for large amounts of intermediate data, such as information from joins. When you implement temporary tables, business intelligence tools can retrieve this data from the temporary table instead of the SQL data service. This results in an increase in performance.

Temporary tables also provide increased security in two ways. First, only the user of the active session can access the tables. Also, the tables persist while a session is active, and the database drops the tables when the connection closes.

To implement temporary tables, an administrator must create the Data Integration Service, and then configure the **Table Storage Connection** in the SQL Connection properties of the Data Integration Service. After the administrator configures the connection, a developer uses the Informatica ODBC or JDBC driver to

configure a connection between a business intelligence tool and the Informatica SQL data service. When these connections are configured, the business intelligence tool can create and use temporary tables.

Temporary tables for all SQL data services in a Data Integration Service use the same relational database connection. When the connection to the SQL data service is active, you can connect to the SQL data service through a JDBC or ODBC client. The relational database drops temporary tables when the session ends. If the Data Integration Service unexpectedly shuts down, the relational database drops temporary tables on the next Data Integration Service startup.

## Temporary Table Implementation

You can store intermediate query result set data in temporary tables when complex queries produce large amounts of intermediate data. For example, temporary tables can store frequently used join results. Business intelligence tools can query the temporary table instead of the SQL data service, resulting in increased performance.

To implement temporary tables, the Informatica administrator and the business intelligence tool user perform the following separate tasks:

**Step 1. The Informatica administrator creates a connection for the data integration service.**

In the Administrator tool, create a connection to the SQL data service. Edit the **SQL Properties** of the Data Integration Service and select a relational database connection for the **Table Storage Connection** property. Recycle the Data Information Service.

**Step 2. The business intelligence tool user creates a connection for the SQL data service.**

In a business intelligence tool, create a connection to the SQL data service. The connection uses the Informatica ODBC or JDBC driver.

**Step 3. Queries from the business intelligence tool create and use temporary tables.**

While the connection is active, the business intelligence tool issues queries to the SQL data service. These queries create and use temporary tables to store large amounts of data that the complex query produces. When the connection ends, the database drops the temporary table.

## Temporary Table Operations

After you create the SQL data service connection, you can use SQL operations to create, populate, select from, or drop a temporary table. You can issue these commands in a regular or stored SQL statement.

You can perform the following operations:

**Create a temporary table.**

To create a temporary table on the relational database, use the following syntax:

```
CREATE TABLE emp (empID INTEGER PRIMARY KEY,eName char(50) NOT NULL,)
```

You can specify the table name in the SQL data service.

**Note:** Use `CREATE TABLE`, not `CREATE TEMPORARY TABLE`. The use of `CREATE TEMPORARY TABLE` is not supported.

**Create a temporary table from a source table.**

You can create a temporary table with or without data from a source table.

The following syntax is supported in Informatica Data Services version 9.5.1:

```
CREATE TABLE emp.backup as select * from emp
```

Where `emp` is an existing schema in the SQL data service that you connected to.

The following syntax is supported in Informatica Data Services version 9.6.0 and 9.6.1:

```
CREATE TABLE emp.backup as select * from emp [ [LIMIT n] ]
```

Where `emp` is an existing schema in the SQL data service that you connected to.

When you create a temporary table with data, the Data Integration Service populates the table with the data. The `CREATE AS` operator copies columns from a database table into the temporary table.

You cannot maintain foreign key or primary key constraints when you use `CREATE AS`.

You can cancel a request before the Data Integration Service copies all the data.

**Note:** The Informatica administrator must create a connection, and then configure it in **SQL Properties** as the **Table Storage Connection**, before you create the temporary table.

#### Insert data into a temporary table.

To insert data into a temporary table, use the `INSERT INTO <temp_table>` statement. You can insert literal data and query data into a temporary table.

The following table shows examples of SQL statements that you can use to insert literal data and query data into a temporary table:

Type	Description
Literal data	<p>Literals describe a user or system-supplied string or value that is not an identifier or keyword. Use strings, numbers, dates, or boolean values when you insert literal data into a temporary table. Use the following statement format to insert literal data into a temporary table:</p> <pre>INSERT INTO &lt;TABLENAME&gt; &lt;OPTIONAL COLUMN LIST&gt; VALUES (&lt;VALUE LIST&gt;), (&lt;VALUE LIST&gt;)</pre> <p>For example, <code>INSERT INTO temp_dept (dept_id, dept_name, location) VALUES (2, 'Marketing', 'Los Angeles')</code>.</p>
Query data	<p>You can query an SQL data service and insert data from the query into a temporary table. Use the following statement format to insert query data into a temporary table:</p> <pre>INSERT INTO &lt;TABLENAME&gt; &lt;OPTIONAL COLUMN LIST&gt; &lt;SELECT QUERY&gt;</pre> <p>For example, <code>INSERT INTO temp_dept(dept_id, dept_name, location) SELECT dept_id, dept_name, location from dept where dept_id = 99</code>.</p> <p>You can use a set operator, such as <code>UNION</code>, in the SQL statement when you insert query data into a temporary table. Use the following statement format when you use a set operator:</p> <pre>INSERT INTO &lt;TABLENAME&gt; &lt;OPTIONAL COLUMN LIST&gt; (&lt;SELECT QUERY&gt; &lt;SET OPERATOR&gt; &lt;SELECT QUERY&gt;)</pre> <p>For example, <code>INSERT INTO temp_dept select * from north_america_dept UNION select * from asia_dept</code>.</p>

#### Select from a temporary table.

You can query the temporary table with the `SELECT ... from <table>` statement.

#### Drop a temporary table.

To drop a temporary table from the relational database, use the following syntax:

```
DROP TABLE <tableName>
```

If the table is not dropped on the physical database, the SQL data service drops the table the next time the Data Integration Service starts, if the table still exists.

## Rules and Guidelines for Temporary Tables

Consider the following rules and guidelines for creation and use of temporary tables:

- You can specify schema and default schema for a temporary table.
- You can place the primary key, NULL, NOT NULL, and DEFAULT constraints on a temporary table.
- You cannot place a foreign key or CHECK and UNIQUE constraints on a temporary table.
- You cannot issue a query that contains a common table expression or a correlated subquery against a temporary table.
- `CREATE AS` statements cannot contain a correlated subquery.

## Virtual Table Mappings

A virtual table mapping defines the virtual data flow between sources and a virtual table in an SQL data service. Use a virtual table mapping to transform the data.

Create a virtual table mapping to link a virtual table in an SQL data service with source data and to define the rules for data transformation. When an end user queries the virtual table, the Data Integration Service applies the transformation rules defined in the virtual table mapping to the source data. It returns the transformed data to the end user.

If you do not want to transform the data, you do not have to create a virtual table mapping. When an end user queries the virtual table, the Data Integration Service retrieves data directly from the data object.

You can create one virtual table mapping for each virtual table in an SQL data service. You can preview virtual table data as you create and update the mapping.

A virtual table mapping contains the following components:

- Sources. Physical or logical data objects that describe the characteristics of source tables or files. A virtual table mapping must contain at least one source.
- Transformations. Objects that define the rules for data transformation. Use different transformation objects to perform different functions. Transformations are optional in a virtual table mapping.
- Virtual table. A virtual table in an SQL data service.
- Links. Connections between columns that define virtual data flow between sources, transformations, and the virtual table.

### Example

You want to make order information available to one of your customers.

The orders information is stored in a relational database table that contains information for several customers. The customer is not authorized to view the orders information for other customers.

Create an SQL data service to retrieve the orders information. Create a virtual table from the orders table and set the data access method to custom. Add a Filter transformation to the virtual table mapping to remove orders data for the other customers.

After you create and deploy an application that contains the SQL data service, the customer can query the virtual table that contains his orders information.

## Defining a Virtual Table Mapping

To define a virtual table mapping, create a virtual table mapping, add sources and transformations, and validate the mapping.

1. Create a mapping from a virtual table in an SQL data service.
2. Add sources and transformations to the mapping and link columns.
3. Validate the mapping.
4. Optionally, preview the mapping data.

## Creating a Virtual Table Mapping

Create a virtual table mapping to define the virtual data flow between source data and a virtual table in an SQL data service. You can create one virtual table mapping for each virtual table.

1. Open the SQL data service that contains the virtual table for which you want to create a virtual table mapping.
2. Click the **Overview** view.
3. In the **Tables** section, change the data access method for the virtual table to **Custom**.  
The **New Virtual Table Mapping** dialog box appears.

4. Enter a name for the virtual table mapping.
5. Click **Finish**.

The Developer tool creates a view for the virtual table mapping and places the virtual table in the editor. If you created the virtual table from a data object, the Developer tool adds the data object to the mapping as a source.

6. To add sources to the mapping, drag data objects from the **Object Explorer** view into the editor.  
You can add logical or physical data objects as sources.
7. Optionally, add transformations to the mapping by dragging them from the **Object Explorer** view or Transformation palette into the editor.
8. Link columns by selecting a column in a source or transformation and dragging it to a column in another transformation or the virtual table.

The Developer tool uses an arrow to indicate the columns are linked.

## Validating a Virtual Table Mapping

Validate a virtual table mapping to verify that the Data Integration Service can read and process the entire virtual table mapping.

1. Open an SQL data service.
2. Select the virtual table mapping view.
3. Select **Edit > Validate**.

The **Validation Log** view opens. If no errors appear in the view, the virtual table mapping is valid.

4. If the **Validation Log** view lists errors, correct the errors and revalidate the virtual table mapping.

## Previewing Virtual Table Mapping Output

As you develop a virtual table mapping, preview the output to verify the virtual table mapping produces the results you want.

The virtual table must be linked to source data.

1. Open the SQL data service that contains the virtual table mapping.
2. Click the virtual table mapping view.
3. Select the object for which you want to preview output. You can select a transformation or the virtual table.
4. Click the **Data Viewer** view.
5. Click **Run**.

The Developer tool displays results in the **Output** section.

## Virtual Stored Procedures

A virtual stored procedure is a set of procedural or data flow instructions in an SQL data service. When you deploy an application that contains an SQL data service, end users can access and run the virtual stored procedures in the SQL data service through a JDBC client tool.

Create a virtual stored procedure to allow end users to perform calculations, retrieve data, or write data to a data object. End users can send data to and receive data from the virtual stored procedure through input and output parameters.

Create a virtual stored procedure within a virtual schema in an SQL data service. You can create multiple stored procedures within a virtual schema.

A virtual stored procedure contains the following components:

- **Inputs.** Objects that pass data into the virtual stored procedure. Inputs can be input parameters, Read transformations, or physical or logical data objects. Input parameters pass data to the stored procedure. Read transformations extract data from logical data objects. A virtual stored procedure must contain at least one input.
- **Transformations.** Objects that define the rules for data transformation. Use different transformation objects to perform different functions. Transformations are optional in a virtual stored procedure.
- **Outputs.** Objects that pass data out of a virtual stored procedure. Outputs can be output parameters, Write transformations, or physical or logical data objects. Output parameters receive data from the stored procedure. Write transformations write data to logical data objects. A virtual stored procedure must contain at least one output. Virtual stored procedures do not return result sets.
- **Links.** Connections between ports that define virtual data flow between inputs, transformations, and outputs.

### Example

An end user needs to update customer email addresses for customer records stored in multiple relational databases.

To allow the end user to update the email addresses, first create a logical data object model to define a unified view of the customer. Create a logical data object that represents a union of the relational tables. Create a logical data object write mapping to write to the relational tables. Add a Router transformation to determine which relational table contains the customer record the end user needs to update.

Next, create an SQL data service. In the SQL data service, create a virtual stored procedure that contains input parameters for the customer ID and email address. Create a Write transformation based on the logical data object and add it to the virtual stored procedure as output.

Finally, deploy the SQL data service. The end user can call the virtual stored procedure through a third-party client tool. The end user passes the customer ID and updated email address to the virtual stored procedure. The virtual stored procedure uses the Write transformation to update the logical data object. The logical data object write mapping determines which relational table to update based on the customer ID and updates the customer email address in the correct table.

## Defining a Virtual Stored Procedure

To define a virtual stored procedure, create a virtual stored procedure, add inputs, transformations, and outputs, and validate the stored procedure.

1. Create a virtual stored procedure in an SQL data service.
2. Add inputs, transformations, and outputs to the virtual stored procedure, and link the ports.
3. Validate the virtual stored procedure.
4. Optionally, preview the virtual stored procedure output.

## Creating a Virtual Stored Procedure

Create a virtual stored procedure to allow an end user to access the business logic within the procedure through a JDBC or ODBC client tool. You must create a virtual stored procedure within a virtual schema.

1. In the **Outline** view for an SQL data service, right-click the data service and select **New > Virtual Stored Procedure**.  
The **New Virtual Stored Procedure** dialog box appears.
2. Enter a name for the virtual stored procedure.
3. Enter a virtual schema name or select a virtual schema.
4. If the virtual stored procedure has input parameters or output parameters, select the appropriate option.
5. Click **Finish**.

The Developer tool opens the virtual stored procedure in the editor. If you select input parameters or output parameters, the Developer tool adds an Input Parameter transformation or an Output Parameter transformation, or both, in the editor.

6. Add input parameters or sources to the virtual stored procedure.
7. Add output parameters or targets to the virtual stored procedure.
8. Optionally, add transformations to the virtual stored procedure by dragging them from the **Object Explorer** view or the Transformation palette into the editor.
9. Link ports by selecting a port in a source or transformation and dragging it to a port in another transformation or target.

The Developer tool uses an arrow to indicate the ports are linked.

## Validating a Virtual Stored Procedure

Validate a virtual stored procedure to verify that the Data Integration Service can read and process the virtual stored procedure.

1. Open a virtual stored procedure.
2. Select **Edit > Validate**.

The **Validation Log** view opens. If no errors appear in the view, the virtual stored procedure is valid.

3. If the **Validation Log** view lists errors, correct the errors and revalidate the virtual stored procedure.

## Previewing Virtual Stored Procedure Output

As you develop a virtual stored procedure, preview the output to verify that the virtual stored procedure produces the results you want.

The virtual stored procedure must contain at least one input parameter or source.

1. Open a virtual stored procedure.
2. Select the **Data Viewer** view.
3. If the virtual stored procedure contains input parameters, enter them in the **Input** section.
4. Click **Run**.

The Developer tool displays results in the **Output** section.

## SQL Query Plans

Use an SQL query plan to view a mapping-like representation of the SQL query that you enter when you preview virtual table data. You can view the original query and the optimized query that the Data Integration Service runs.

When you view the SQL query plan for a query, the Developer tool displays a graphical representation of the query that looks like a mapping. The graphical representation has a source, transformations, links, and a target. View the query plan to troubleshoot queries that end users run against a deployed SQL data service and understand log messages.

## SQL Query Plan Optimization

The Data Integration Services optimizes an SQL query to increase performance when you select a optimizer level. The optimized query produces the same results but runs more quickly.

The Data Integration Service can perform the following type of optimizations:

- Push transformations to relational data objects.
- Reorder transformations in the mapping.
- Push SQL set operations such as UNION, UNION ALL, DISTINCT, INTERSECT, DISTINCT, and MINUS to relational data objects.
- Push SQL keyword LIMIT to IBM DB2, MS SQL, and Oracle relational data objects.

You can view the original query plan and the optimized query plan from the **Data Viewer** view. The resulting optimized query can contain different transformations or transformations in a different order. The Data



Integration Service can push transformations and SQL operations to the relational data object to minimize data read from the source.

You can configure different optimizer levels in the Developer tool. Different optimizer levels produce different queries. The query optimization depends on the optimizer level that you select and the complexity of the query. When you run a simple query against a virtual table, different optimizer levels might produce the same optimized query. When you run a query that contains multiple clauses and subqueries, different optimizer levels produce a different optimized queries.

## SQL Query Plan Optimization Examples

The Data Integration Service can optimize SQL operations and transformations by pushing them to a relational data object. For example, you can query a customer virtual table to select distinct customers or filter customer data by customer ID.

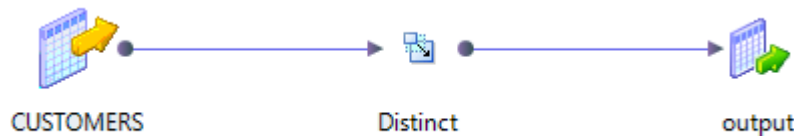
### DISTINCT SQL Operation

You want to query the CUSTOMERS virtual table in an SQL data service to select distinct customer numbers. The Data Integration Service can push SQL keywords such as Distinct to the relational data object with the normal optimizer level.

You can enter the following query in the **Data Viewer** view:

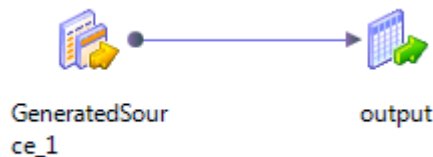
```
select distinct Customer_number from CUSTOMERS
```

The following figure shows the SQL query plan that appears in the **Non-Optimized** tab:



The non-optimized representation displays the query plan based on the query you enter with the DISTINCT operation.

The following figure shows the SQL query plan that appears in the **Optimized** tab:



The optimized representation displays the query plan as the Data Integration Service runs it. The Data Integration Service pushes the DISTINCT operation to the source to increase performance.

### Filter Transformation

You want to query the CUSTOMERS virtual table in an SQL data service to filter and order customer data. The Data Integration Service can push transformations such as a Filter transformation to the relational data object with the normal optimizer level.

You can enter the following query in the **Data Viewer** view:

```
select * from CUSTOMERS where CUSTOMER_ID > 150000 order by LAST_NAME
```

The following figure shows the SQL query plan that appears in the **Non-Optimized** tab:



The non-optimized representation displays the query plan based on the query you enter. The Developer tool displays the WHERE clause as a Filter transformation and the ORDER BY clause as a Sorter transformation. The Developer tool uses a pass-through Expression transformation to rename ports.

The following figure shows the optimized SQL query plan that appears in the **Optimized** tab:



The optimized representation displays the query plan as the Data Integration Service runs it. Because the optimizer level is normal, the Data Integration Service pushes the filter condition to the source. Pushing the filter condition improves query performance because it reduces the number of rows that the Data Integration Service reads from the source.

As in the non-optimized query, the Developer tool displays the ORDER BY clause as a Sorter transformation. The Data Integration Service uses pass-through Expression transformations to enforce the data types that you configure in the logical transformations.

## Viewing an SQL Query Plan

Display the SQL query plan to view a mapping-like representation of the SQL query you enter when you preview virtual table data.

1. Open an SQL data service that contains at least one virtual table.
2. Click the **Data Viewer** view.
3. Enter an SQL query in the **Input** window.
4. Optionally, select a data viewer configuration that contains the optimizer level you want to apply to the query.
5. Click **Show Query Plan**.

The Developer tool displays the SQL query plan for the query as you entered it on the **Non-Optimized** tab.

6. To view the optimized query, click the **Optimized** tab.

The Developer tool displays the optimized SQL query plan.

## CHAPTER 3

# SQL Syntax

This chapter includes the following topics:

- [SQL Syntax Overview, 27](#)
- [Datatypes, 28](#)
- [Operators, 28](#)
- [Functions, 28](#)
- [SQL Statements and Keywords, 36](#)
- [Queries, 37](#)
- [Reserved Words, 39](#)
- [Escape Syntax, 39](#)
- [Troubleshooting SQL Data Services, 42](#)

## SQL Syntax Overview

An SQL data service is a virtual database you can query. It provides a uniform view of data that might be scattered among disparate, heterogeneous data sources. You can run SQL queries when you preview virtual table data in Developer tool. You can also use a JDBC or ODBC client tool to run SQL queries against the virtual tables in an SQL data service.

An SQL data service can contain the following objects:

- Virtual schemas that define the database structure
- Virtual tables
- Virtual table mappings that define the flow of data between sources and a virtual table
- Virtual stored procedures that take optional inputs through parameters, transform the data, and optionally return output through parameters

To allow end users to query the virtual tables and run the virtual stored procedures in an SQL data service, a developer creates and deploys an application that contains the SQL data service. The developer deploys the application to a Data Integration Service and an administrator runs the application. When the application is running, end users can make SQL queries against the virtual tables and run the virtual stored procedures in the SQL data service.

SQL data services support ANSI SQL-92 operators, functions, statements, and keywords.

# Datatypes

SQL data services support common SQL datatypes.

SQL data services support the following datatypes:

- Bigint
- Binary
- Boolean
- Char
- Date
- Decimal
- Double
- Int
- Time
- Timestamp
- Varchar

# Operators

SQL data services support common operators. Use operators to perform mathematical computations, combine data, or compare data.

SQL data services support the following operators in an SQL query:

- Arithmetic operators: (), unary + and -, \*, /, +, -
- Comparison operators: =, !=, <>, <, <=, >, >=
- Logical operators: AND, NOT, OR
- || (concatenate strings)
- BETWEEN
- CASE
- EXISTS
- IN, NOT IN
- IS NULL, IS NOT NULL
- LIKE, NOT LIKE

# Functions

You can use SQL and Informatica functions to run queries against an SQL data service.

Some functions are SQL and Informatica equivalents, such as the ABS function. Some functions are unique to ANSI SQL or to Informatica.

**Note:** You cannot use filter conditions with Informatica functions in the SQL.

The following table provides the syntax and functions that you can use to query an SQL data service:

Function	Syntax	Description
ABS	ABS( numeric_value )	Returns the absolute value of a numeric value. Informatica and SQL function.
ADD_TO_DATE	ADD_TO_DATE( date, format, amount )	Adds a specified amount to one part of a datetime value, and returns a date in the same format as the date you pass to the function. Informatica and SQL function.
ASCII	ASCII ( string )	Returns the numeric ASCII or Unicode value of the first character of the string passed to the function. Informatica and SQL function.
AVG	AVG( numeric_value )	Returns the average of all values in a group of rows. SQL function.
CASE (Simple)	CASE input_expression WHEN when_expression THEN result_expression [ ...n ] [ ELSE else_result_expression ] END	Compares an expression to a set of simple expressions and returns a result associated with the first matching value. SQL function.
CASE (Searched)	CASE WHEN Boolean_expression THEN result_expression [ ...n ] [ ELSE else_result_expression ] END	Evaluates a set of Boolean expressions and returns the first true result. SQL function.
CEIL	CEIL( numeric_value )	Returns the smallest integer greater than or equal to the numeric value passed to this function. Informatica and SQL function.
CHAR_LENGTH	CHAR_LENGTH( numeric_value )	Returns the number of characters in a string, including trailing blanks. SQL function.
CHR	CHR( numeric_value )	Returns the ASCII or Unicode character that corresponds to the numeric value you pass to this function. Informatica and SQL function.
CHRCODE	CHRCODE ( string )	Returns the numeric ASCII or Unicode value of the first character of the string passed to the function. SQL function.

Function	Syntax	Description
COALESCE	COALESCE ( first_argument, second_argument[, third_argument, ...] )	Returns the first non-NULL argument from a list of arguments. If all arguments are NULL, COALESCE returns NULL. SQL function.
CONCAT	CONCAT( first_string, second_string )	Concatenates two strings. Informatica and SQL function.
CONVERT_BASE	CONVERT_BASE( string, source_base, dest_base )	Converts a non-negative numeric string from one base value to another base value. Informatica and SQL function.
COS	COS( numeric_value )	Returns the cosine, expressed in radians, of a numeric value. Informatica and SQL function.
COSH	COSH( numeric_value )	Returns the hyperbolic cosine, expressed in radians, of a numeric value. Informatica and SQL function.
COUNT	COUNT( value )	Returns the number of rows that have non-null values in a group. Optionally, you can include the asterisk (*) argument to count all input values in a transformation. SQL function.
CRC32	CRC32( value )	Returns a 32-bit Cyclic Redundancy Check (CRC32) value. Informatica and SQL function.
CUME	CUME( numeric_value )	Returns a running total. A running total means CUME returns a total each time it adds a value. SQL function.
CURRENT_DATE	CURRENT_DATE	Returns the current date on the node hosting the Data Integration Service. The format of the returned value depends on the locale of the client machine. The returned value does not change if this function is executed more than once in a statement. SQL function.
CURRENT_TIME	CURRENT_TIME	Returns the current time on the node hosting the Data Integration Service. The format of the returned value depends on the locale of the client machine. The returned value does not change if this function is executed more than once in a statement. SQL function.
CURRENT_TIMESTAMP	CURRENT_TIMESTAMP	Returns the current date and time on the node hosting the Data Integration Service. The format of the returned value depends on the locale of the client machine. The returned value does not change if this function is executed more than once in a statement. SQL function.

Function	Syntax	Description
DATE_COMPARE	DATE_COMPARE( date1, date2 )	Returns an integer that indicates which of two dates is earlier: <ul style="list-style-type: none"> <li>- -1 if the first date is earlier</li> <li>- 0 if the two dates are equal</li> <li>- 1 if the second date is earlier</li> <li>- NULL if one of the date values is NULL</li> </ul> Informatica and SQL function.
DATE_DIFF	DATE_DIFF( date1, date2, format )	Returns the length of time between two dates. Informatica and SQL function.
EXP	EXP( exponent )	Returns e raised to the specified power (exponent), where e=2.71828183. Informatica and SQL function.
EXTRACT	EXTRACT( YEAR MONTH DAY  HOUR MINUTE SECOND FROM date )	Extracts the specified part of a date as an integer value. Therefore, if you create an expression that returns the day portion of the date, and pass a date such as Apr 15 2007 00:00:00, EXTRACT returns 15. SQL function.
FLOOR	FLOOR( numeric_value )	Returns the largest integer less than or equal to the numeric value you pass to this function. Informatica and SQL function.
FV	FV( rate, terms, payment [, present value, type] )	Returns the future value of an investment, where you make periodic, constant payments, and the investment earns a constant interest rate. Informatica and SQL function.
GET_DATE_PART	GET_DATE_PART( date, format )	Returns the specified part of a date as an integer value. Therefore, if you create an expression that returns the month portion of the date, and pass a date such as Apr 1 1997 00:00:00, GET_DATE_PART returns 4. Informatica and SQL function.
INITCAP	INITCAP( string )	Capitalizes the first letter in each word of a string and converts all other letters to lowercase. Informatica and SQL function.
INSTR	INSTR( string, search_value [,start [,occurrence [,comparison_type ]]] )	Returns the position of a character set in a string, counting from left to right. Informatica and SQL function.
IS_DATE	IS_DATE( value [,format] )	Returns whether a string value is a valid date. Informatica and SQL function.
IS_NUMBER	IS_NUMBER( value )	Returns whether a string is a valid number. Informatica function.

Function	Syntax	Description
IS_SPACES	IS_SPACES( value )	Returns whether a string value consists entirely of spaces. A space is a blank space, a formfeed, a newline, a carriage return, a tab, or a vertical tab. Informatica function.
ISNULL	ISNULL( value )	Returns whether a value is NULL. Evaluates an empty string as FALSE. Informatica and SQL function.
ISNUMERIC	ISNUMERIC( value )	Returns whether a string is a valid number. SQL function.
LAST_DAY	LAST_DAY( date )	Returns the date of the last day of the month for each date in a column. Informatica and SQL function.
LN	LN( numeric_value )	Returns the natural logarithm of a numeric value. Informatica and SQL function.
LOCATE	LOCATE( string, search_value )	Returns the position of a character set from the beginning of a string. SQL function.
LOG	LOG( base, exponent )	Returns the logarithm of a numeric value. Informatica and SQL function.
LOWER	LOWER( string )	Converts uppercase string characters to lowercase. Informatica and SQL function.
LPAD	LPAD( first_string, length [,second_string] )	Adds a set of blanks or characters to the beginning of a string to set the string to a specified length. Informatica and SQL function.
LTRIM	LTRIM( string [, trim_set] )	Removes blanks or characters from the beginning of a string. Informatica and SQL function.
MAKE_DATE_TIME	MAKE_DATE_TIME( year, month, day, hour, minute, second, nanosecond )	Returns the date and time based on the input values. Informatica and SQL function.
MAX	MAX( value )	Returns the latest date, maximum numeric value, or highest string value found within a column or group. SQL function.
MD5	MD5( value )	Calculates the checksum of the input value. Informatica and SQL function.
METAPHONE	METAPHONE( string [,length] )	Encodes string values. You can specify the length of the string that you want to encode. Informatica and SQL function.



Function	Syntax	Description
MIN	MIN( value )	Returns the earliest date, minimum numeric value, or lowest string value found within a column or group. SQL function.
MOD	MOD( numeric_value, divisor )	Returns the remainder of a division calculation. Informatica and SQL function.
MOVINGAVG	MOVINGAVG( numeric_value, rowset )	Returns the row-by-row average of a specified set of rows. Informatica function.
MOVINGSUM	MOVINGSUM( numeric_value, rowset )	Returns the row-by-row sum of a specified set of rows. Informatica function.
NPER	NPER( rate, present value, payment [, future value, type] )	Returns the number of periods for an investment based on a constant interest rate and periodic, constant payments. Informatica and SQL function.
PMT	PMT( rate, terms, present value [, future value, type] )	Returns the payment for a loan based on constant payments and a constant interest rate. Informatica and SQL function.
POSITION	POSITION( search_value, string )	Returns the position of a character set from the beginning of a string. SQL function.
POWER	POWER( base, exponent )	Returns a value raised to the exponent you pass to the function. Informatica and SQL function.
PV	PV( rate, terms, payment [, future value, type] )	Returns the present value of an investment. Informatica and SQL function.
RAND	RAND( seed )	Returns a random number between 0 and 1. For the same seed, the Data Integration Service generates the same sequence of numbers. Informatica and SQL function.
RATE	RATE( terms, payment, present value[, future value, type] )	Returns the interest rate earned per period by a security. Returns NULL if present value is greater than the product of terms and payment. Informatica and SQL function.
REG_EXTRACT	REG_EXTRACT( subject, 'pattern', subPatternNum )	Extracts subpatterns of a regular expression within an input value. For example, from a regular expression pattern for a full name, you can extract the first name or last name. Informatica function.

Function	Syntax	Description
REG_MATCH	REG_MATCH( subject, pattern )	Returns whether a value matches a regular expression pattern to validate data patterns, such as IDs, telephone numbers, postal codes, and state names. Informatica function.
REG_REPLACE	REG_REPLACE( subject, pattern, replace, numReplacements )	Replaces characters in a string with another character pattern. Searches the input string for the character pattern you specify and replaces all occurrences with the replacement pattern. You can also indicate the number of occurrences of the pattern you want to replace in the string. Informatica function.
REPLACECHR	REPLACECHR( CaseFlag, InputString, OldCharSet, NewChar )	Replaces characters in a string with a single character or no character. Informatica and SQL function.
ROUND (dates)	ROUND( date [,format] )	Rounds one part of a date. Informatica and SQL function.
ROUND (numbers)	ROUND( numeric_value [, precision] )	Rounds numbers to a specified number of digits or decimal places. Informatica and SQL function.
RPAD	RPAD( first_string, length [,second_string] )	Converts a string to a specified length by adding blanks or characters to the end of the string. Informatica function.
RTRIM	RTRIM( string [, trim_set] )	Removes blanks or characters from the end of a string. Informatica and SQL function.
SET_DATE_PART	SET_DATE_PART( date, format, value )	Sets one part of a Date/Time value to a value you specify. Informatica and SQL function.
SIGN	SIGN( numeric_value )	Returns whether a numeric value is positive, negative, or 0. Informatica and SQL function.
SIN	SIN( numeric_value )	Returns the sine, expressed in radians, of a numeric value. Informatica and SQL function.
SINH	SINH( numeric_value )	Returns the hyperbolic sine, expressed in radians, of a numeric value. Informatica and SQL function.

Function	Syntax	Description
SOUNDEX	SOUNDEX( string )	<p>Encodes a string value into a four-character string. It uses the first character of the input string as the first character in the return value and encodes the remaining three unique consonants as numbers:</p> <ul style="list-style-type: none"> <li>- B, F, P, V = 1</li> <li>- C, G, J, K, Q, S, X, Z = 2</li> <li>- D, T = 3</li> <li>- L = 4</li> <li>- M, N = 5</li> <li>- R = 6</li> </ul> <p>Informatica and SQL function.</p>
SQRT	SQRT( numeric_value )	<p>Returns the square root of a non-negative numeric value.</p> <p>Informatica and SQL function.</p>
SUBSTR	SUBSTR( string, start [,length] )	<p>Returns a portion of a string. Counts all characters, including blanks.</p> <p>Informatica and SQL function.</p>
SUM	SUM( numeric_value )	<p>Returns the sum of all values in the selected column.</p> <p>SQL function.</p>
TAN	TAN( numeric_value )	<p>Returns the tangent, expressed in radians, of a numeric value.</p> <p>Informatica and SQL function.</p>
TANH	TANH( numeric_value )	<p>Returns the hyperbolic tangent, expressed in radians, of a numeric value.</p> <p>Informatica and SQL function.</p>
TO_BIGINT	TO_BIGINT( value [, flag] )	<p>Converts a string or numeric value to a bigint value. The flag truncates the decimal portion if true or nonzero. It rounds the decimal portion if false or 0.</p> <p>Informatica and SQL function.</p>
TO_CHAR	TO_CHAR( value )	<p>Converts numeric values or dates to text strings.</p> <p>Informatica and SQL function.</p>
TO_DATE	TO_DATE( string [, format] )	<p>Converts a character string to a Date/Time datatype.</p> <p>Informatica and SQL function.</p>
TO_DECIMAL	TO_DECIMAL( value [, scale] )	<p>Converts a string or numeric value to a decimal value.</p> <p>Informatica and SQL function.</p>
TO_FLOAT	TO_FLOAT( value )	<p>Converts a string or numeric value to a double-precision floating point number (the Double datatype).</p> <p>Informatica and SQL function.</p>

Function	Syntax	Description
TO_INTEGER	TO_INTEGER( value [, flag] )	Converts a string or numeric value to an integer. The flag truncates the decimal portion if true or nonzero. It rounds the decimal portion if false or 0. Informatica and SQL function.
TRIM	TRIM( [operand] string )	Removes leading characters, trailing characters, or both from a string. Operands are as follows: - LEADING [ trim_character FROM ] - TRAILING [ trim_character FROM ] - BOTH [ trim_character FROM ] The trim_character argument passes the character you want to remove. Enclose the trim character in single quotes. If you omit the trim character, TRIM removes spaces. For example, TRIM( ' Mary ' ) returns Mary (no spaces). TRIM( BOTH '*' FROM '**Mary*' ) returns Mary. SQL function.
TRUNC (dates)	TRUNC( date [,format] )	Truncates dates to a specific year, month, day, hour, minute, second, millisecond, or microsecond. Informatica and SQL function.
TRUNC (numbers)	TRUNC( numeric_value [, precision] )	Truncates numbers to a specific digit. Informatica and SQL function.
UPPER	UPPER( string )	Converts lowercase string characters to uppercase. Informatica and SQL function.

## SQL Statements and Keywords

SQL data services support the ANSI SQL-92 standard. Use SQL statements and keywords to define SQL queries and run virtual stored procedures.

To query the virtual tables in an SQL data service, use the SQL SELECT statement.

To run the virtual stored procedures in an SQL data service, use the SQL CALL (PROCEDURE) statement. The syntax is as follows:

```
CALL proc_name( [parameter_1] [, parameter_n] )
```

You can use the following clauses in an SQL query:

- FROM
- GROUP BY
- HAVING
- ORDER BY
- WHERE

You can use the following SQL keywords in an SQL query:

- ALL
- CROSS JOIN
- DISTINCT
- EXCEPT
- FULL OUTER JOIN
- INNER JOIN
- INTERSECT
- LEFT OUTER JOIN
- LIMIT
- MINUS
- RIGHT OUTER JOIN
- UNION, UNION ALL

## Queries

You can issue non-correlated subqueries, correlated subqueries, and parameterized queries when you query virtual tables and run virtual stored procedures in an SQL data service.

### Non-Correlated Subqueries

A non-correlated subquery is a subquery that is not dependent on the outer query. Use non-correlated subqueries to filter or modify data when you query virtual tables in an SQL data service.

You can use non-correlated subqueries in the following places:

- Expressions
- BETWEEN operator
- CASE operator
- FROM clause
- HAVING clause
- IN, NOT IN operators
- SELECT statement
- WHERE clause

You can use scalar non-correlated subqueries.

You can use non-correlated subqueries as SET operands.

## Correlated Subqueries

A correlated subquery is a subquery that uses values from the outer query in its `WHERE` clause. The subquery is evaluated once for each row processed by the outer query. Use correlated subqueries to filter or modify data when you query virtual tables in an SQL data service.

You can issue a correlated subquery from an ODBC client, JDBC client, or from the query plan window in the Developer tool.

The following table provides examples of the types of correlated subqueries that you can issue against an SQL data service:

Type	Description
IN	<p>A correlated subquery that uses the <code>IN</code> keyword within an SQL <code>WHERE</code> clause to select rows from the values returned by the correlated subquery.</p> <p>For example, <code>SELECT * FROM vs.nation a WHERE a.n_regionkey IN (SELECT distinct b.r_regionkey FROM vs.region b WHERE b.r_regionkey = a.n_regionkey)</code>.</p>
Quantified comparison	<p>A correlated subquery that contains a comparison operator within an SQL <code>WHERE</code> clause.</p> <p>For example, <code>SELECT n_name FROM vs.nation a WHERE 2 &gt; (SELECT 1 FROM vs.nation b WHERE a.n_nationkey=b.n_nationkey)</code>.</p>

## Correlated Subquery Requirements

The Data Integration Service flattens the correlated subqueries into a normal join before it runs the query. If the Data Integration Service cannot flatten a correlated query into a normal join, the Data Integration Service generates a system error and it does not run the query.

The following table shows the results of a correlated subquery that the Data Integration Service flattened:

Type	Query
Non-flattened	<code>SELECT huge.* FROM huge WHERE c1 IN (SELECT c1 FROM tiny)</code>
Flattened	<code>SELECT huge.* FROM huge, tiny WHERE huge.c1 = tiny.c1</code>

The Data Integration Service can flatten a correlated subquery into a normal join when it meets the following requirements:

- The type is `IN` or a quantified comparison.
- It is not within an `OR` operator or part of a `SELECT` list.
- It does not contain the `LIMIT` keyword.
- It does not contain a `GROUP BY` clause, aggregates in a `SELECT` list, or an `EXIST` or `NOT IN` logical operator.
- It generates unique results. One column in the correlated subquery is a primary key. For example, if `r_regionkey` column is a primary key for the `vs.nation` virtual table, you can issue the following query:  
`SELECT * FROM vs.nation WHERE n_regionkey IN (SELECT b.r_regionkey FROM vs.region b WHERE b.r_regionkey = n_regionkey)`.
- If it contains a `FROM` list, each table in the `FROM` list is a virtual table in the SQL data service.

## Parameterized Queries

A parameterized query uses a precompiled SQL statement with placeholders for values that change. Parameterized queries can improve processing efficiency and protect the database from SQL injection attacks. You can use prepared statements and call stored procedures in a parameterized query that you run against an SQL data service.

Define parameters in the `PreparedStatement` or `CallableStatement` object in a JDBC program or in a statement handle prepared by `SQLPrepare` for an ODBC program. Use the `PreparedStatement` object to store a precompiled SQL statement that you can run multiple times. Use the `CallableStatement` object to call stored procedures.

You can use standard method calls and set methods in the `PreparedStatement` object of the parameterized query.

An SQL data service accepts common datatypes when you configure default values for parameters in stored procedures. The date, time, and timestamp datatypes default to the ISO format.

You cannot use the following items in a parameterized query that you run against an SQL data service:

- `addBatch()` and `executeBatch()` prepared statement method calls
- Set methods as of JDBC 6.0
- DDL and DML statements that return update counts
- Stored procedures that return values
- Array datatype

## Reserved Words

Some keywords are reserved for specific functions.

The following words are reserved words:

- All transformation language reserved words
- All ANSI SQL 92-99 keywords

To use reserved words in an SQL query, enclose the word in double quotation marks.

## Escape Syntax

SQL data services support escape clauses for functions, date formats, time formats, and timestamp formats.

An escape clause contains a keyword enclosed in curly brackets.

The following table lists the keywords you can use in an escape clause:

Category	Keyword	Syntax
Functions	fn	<pre>{fn function_name( arguments )}</pre> <p>For example:</p> <pre>SELECT {fn SUBSTRING( EmpName, 1, 2 )} from Emp</pre>
Date formats	d	<pre>{d 'value'}</pre> <p>The format for the date value must match the SQL data service default date format. Therefore, if the default date format for the SQL data service is YYYY-MM-DD, the date value must include a 4-digit year.</p> <p>For example:</p> <pre>SELECT * FROM Orders WHERE OrderDate &gt; {d '2005-01-01'}</pre>
Time formats	t	<pre>{t 'value'}</pre> <p>The format for the time value must match the SQL data service default time format. Therefore, if the default time format for the SQL data service is HH:MI:SS, the time value cannot include fractional seconds.</p> <p>For example:</p> <pre>SELECT * FROM Orders WHERE OrderTime &lt; {t '12:00:00'}</pre>
Timestamp formats	ts	<pre>{ts 'value'}</pre> <p>The format for the timestamp value must match the SQL data service default timestamp format. Therefore, if the default timestamp format for the SQL data service is YYYY-MM-DD HH:MI:SS, the timestamp value cannot include fractional seconds.</p> <p>For example:</p> <pre>SELECT * FROM Sales WHERE TransactTime &gt; {ts '2010-01-15 12:00:00'}</pre>

## Escape Syntax for the fn Keyword

SQL data services support certain functions when you use a function escape sequence.

The following table lists the functions that SQL data services support when you use a function escape sequence:

Function	Syntax	Description
CURTIMESTAMP	<pre>CURTIMESTAMP()</pre>	Returns the current date and time on the node hosting the Data Integration Service. The returned value does not change if it is executed more than once in a single statement.
EXP	<pre>EXP( exponent )</pre>	Returns e raised to the specified power (exponent), where e=2.71828183.
EXTRACT	<pre>EXTRACT( YEAR MONTH DAY          HOUR MINUTE SECOND FROM         date )</pre>	Extracts the specified part of a date as an integer value. Therefore, if you create an expression that returns the day portion of the date, and pass a date such as Apr 15 2007 00:00:00, EXTRACT returns 15.



Function	Syntax	Description
FLOOR	FLOOR( numeric_value )	Returns the largest integer less than or equal to the numeric value you pass to this function.
LCASE	LCASE( string )	Converts uppercase string characters to lowercase.
LENGTH	LENGTH( string )	Returns the number of characters in a string, including trailing blanks.
LOCATE	LOCATE( string, search_value )	Returns the position of a character set from the beginning of a string.
LOG	LOG( numeric_value )	Returns the base-10 logarithm of a numeric value.
LTRIM	LTRIM( string )	Removes blanks from the beginning of a string .
MOD	MOD( numeric_value, divisor )	Returns the remainder of a division calculation.
POWER	POWER( base, exponent )	Returns a value raised to the exponent you pass to the function.
RTRIM	RTRIM( string )	Removes blanks from the end of a string .
SIN	SIN( numeric_value )	Returns the sine, expressed in radians, of a numeric value.
SINH	SINH( numeric_value )	Returns the hyperbolic sine, expressed in radians, of a numeric value.
SQRT	SQRT( numeric_value )	Returns the square root of a non-negative numeric value.
SUBSTRING	SUBSTRING( string, start [,length] )	Returns a portion of a string. Counts all characters, including blanks.
TAN	TAN( numeric_value )	Returns the tangent, expressed in radians, of a numeric value.
TANH	TANH( numeric_value )	Returns the hyperbolic tangent, expressed in radians, of a numeric value.
TRIM	TRIM( [operand] string )	<p>Removes leading characters, trailing characters, or both from a string.</p> <p>Operands are as follows:</p> <ul style="list-style-type: none"> <li>- LEADING [ trim_character FROM ]</li> <li>- TRAILING [ trim_character FROM ]</li> <li>- BOTH [ trim_character FROM ]</li> </ul> <p>The trim_character argument passes the character you want to remove. Enclose the trim character in single quotes. If you omit the trim character, TRIM removes spaces.</p> <p>For example, TRIM( ' Mary ' ) returns Mary (no spaces). TRIM( BOTH '*' FROM '**Mary*' ) returns Mary.</p>
UCASE	UCASE( string )	Converts lowercase string characters to uppercase.

# Troubleshooting SQL Data Services

The solutions to the following situations might help you troubleshoot SQL data services.

## When I use special characters in a query against a virtual table in an SQL data service, the query fails.

If the name of a virtual table or a column contains special characters, you must enclose the name in double quotes. For example, an SQL data service contains a virtual table named "@Customers." To return all rows from the table, enter the following query:

```
SELECT * FROM "@Customers"
```

## When I use the CONVERT\_BASE function to convert a negative number in an SQL query, the query fails.

The CONVERT\_BASE function converts non-negative numeric strings. To convert binary data with negative values, you can enter the following query:

```
CONVERT_BASE(TO_CHAR(X-ABS(value)), source_base, dest_base) )
```

X represents the destination base raised to the number of bits.

For example, the following query uses  $2^{16}=65536$  and calculates the negative integers for binary data with 16 bits:

```
CONVERT_BASE(TO_CHAR(65536-ABS(NEWFIELD)), 10, 2) )
```

## I ran an SQL query and the query failed.

You can get an XML file of the failed query mapping and then send the file to Informatica Global Customer Support for analysis. To create an XML file for the query, enable the dumpMapping parameter in the connection properties. When you run a query with the dumpmapping parameter enabled, the Data Integration Service creates an XML file for the query mapping and stores it in the following location:

```
<Informatica installation directory>\tomcat\bin\dslogs\sql.
```

The following table describes how to enable the dumpMapping parameter for each SQL data service connection type:

Connection Type	Method
JDBC connections	Add the following value to the JDBC connection string: &dumpMapping=true
ODBC connections on Windows	Enter the following value in the <b>Optional Parameters</b> field in the <b>Create a New Data Source</b> window: dumpMapping=true
ODBC connections on UNIX	Add the following value to the <code>odbc.ini</code> file: &dumpMapping=true

## I entered an SQL query that converts a large number to a binary value using the CONVERT\_BASE function, and the result is truncated.

Use the CAST() function when converting large numbers to binary. For example, the following CONVERT\_BASE query converts 2222 from base 10 to base 2:

```
CAST(CONVERT_BASE( 2222, 10, 2 ) AS VARCHAR(100))
```

## I entered an SQL query that converts a large number to a binary value using the CONVERT\_BASE function, and the result is truncated.

Use the CAST() function when converting large numbers to binary. For example, the following CONVERT\_BASE query converts 2222 from base 10 to base 2:

```
CAST(CONVERT_BASE( 2222, 10, 2 ) AS VARCHAR(100))
```

**When I use the TO\_DECIMAL function to convert a string or numeric value to a decimal value, the query fails with a decimal overflow error, or the query returns an unexpected decimal value.**

Use the CAST() function to change the SQL statement when you use the TO\_DECIMAL function in an SQL query. For example, the following TO\_DECIMAL query uses the CAST function to return the decimal value 60.250:

```
CAST(TO_DECIMAL((60 + .25, 3))
```

## CHAPTER 4

# Installing and Configuring Drivers for Third-Party Clients

This chapter includes the following topics:

- [Installing and Configuring Drivers for Third-Party Clients Overview, 44](#)
- [Before You Install the Drivers, 45](#)
- [Install the Drivers on Windows, 47](#)
- [Install the Drivers on UNIX, 48](#)
- [JDBC Connections, 50](#)
- [ODBC Connections, 53](#)

## Installing and Configuring Drivers for Third-Party Clients Overview

You can connect to an SQL data service from third-party client tools. Before you can connect from a third-party client tool, you must install and configure the Informatica JDBC/ODBC drivers on the client machine.

Before you can connect to an SQL data service from a third party client tool, you must perform the following tasks:

1. Record the Informatica domain information.
2. Configure the client for secure communication or Kerberos authentication.
3. Install or upgrade the Informatica JDBC/ODBC drivers.
4. Configure the Informatica JDBC/ODBC drivers.

# Before You Install the Drivers

Before you install the Informatica JDBC/ODBC drivers, you must extract the installer files. You also need to record information about the domain. If you connect to an SQL data service in a secure domain, you must configure the client for secure communication or Kerberos authentication.

## Extract the Installer Files

The installer files are compressed and distributed as a zip file on Windows and as a tar file on UNIX.

Use a zip utility, a native tar utility, or a GNU tar utility to extract the installer files to a directory on your machine.

You can extract the installer files in the following ways:

- Installation DVD. Download the Informatica zip or tar file from the installation DVD to a directory on your machine and then extract the installer files. Or, extract the installer files directly from the DVD to a directory on your machine.
- FTP download. Download the Informatica installation zip or tar file from the Informatica Electronic Software Download site to a directory on your machine and then extract the installer files.

## Record Informatica Domain Information

When you install and configure the drivers to access an SQL data service, you need to know information about the domain. If you do not have the domain information, contact the Informatica domain administrator.

The following table describes information that you need to know about the domain:

Required Information	Description
Data Integration Service name	Data Integration Service that runs the application that contains the SQL data service.
Domain host name	Machine that hosts the Informatica domain.
Domain HTTP port	Informatica domain HTTP port number.
SQL data service name	Name of the SQL data service that contains the virtual tables you want to query or the virtual stored procedures that you want to run. The run-time SQL data service name includes the application name that contains the SQL data service and uses the following format: <application name>.<SQL data service name>
Security domain name	Informatica security domain name. Required if the Informatica user account is in an LDAP security domain.
User name	Informatica domain user name.
User password	Informatica domain user password.

Required Information	Description
Truststore file	If the Informatica domain has secure communication enabled, you must have the location of the truststore file that contains the SSL certificate for the domain.
Authentication type	<p>The mode of authentication used to connect to the SQL data service. You can select one of the following authentication modes:</p> <p><b>Native or LDAP Authentication</b></p> <p>Uses an Informatica domain user account to connect to the SQL data service in an Informatica domain that uses Native or LDAP authentication. The user account can be in a native or LDAP security domain.</p> <p><b>Kerberos with keytab</b></p> <p>Uses the service principal name (SPN) of an Informatica domain user account to connect to the SQL data service in an Informatica domain that uses Kerberos authentication.</p> <p><b>Kerberos with user name and password</b></p> <p>Uses an Informatica domain user account to connect to the SQL data service in an Informatica domain that uses Kerberos authentication.</p> <p><b>Logged in user</b></p> <p>Uses the user account logged in to the client machine to connect to the SQL data service in an Informatica domain that uses Native, LDAP, or Kerberos authentication.</p>

## Configure the Client for Secure Communication

If you connect to an SQL data service in an Informatica domain that has secure communication enabled, you must copy the truststore files to the machine that hosts the JDBC or ODBC client and set the `INFA_TRUSTSTORE` and `INFA_TRUSTSTORE_PASSWORD` environment variables.

if you use the default Informatica SSL certificate or if you specify the SSL certificates to use, copy the truststore files and set the environment variables.

Set the following environment variables for the truststore information:

### **INFA\_TRUSTSTORE**

Set this variable to the directory that contains the truststore files for the SSL certificates. The directory must contain truststore files named `infa_truststore.jks` and `infa_truststore.pem`.

### **INFA\_TRUSTSTORE\_PASSWORD**

Set this variable to the password for the `infa_truststore.jks` file. The password must be encrypted. Use the command line program `mpasswd` to encrypt the password.

Set the environment variables based on the SSL certificates that you use for the domain.

If you provide the SSL certificates to use, copy the truststore files to the machine that hosts the client and set the `INFA_TRUSTSTORE` variable to the directory that contains the truststore files. You must have truststore files in JKS and PEM format named `infa_truststore.jks` and `infa_truststore.pem`. You must also set the `INFA_TRUSTSTORE_PASSWORD` variable with the password for the `infa_truststore.jks` file.

## Configure the Client for Kerberos Authentication

If you connect to an SQL data service in an Informatica domain that runs on a network that uses Kerberos authentication, you must copy the Kerberos configuration file to the machine the hosts the client. You might need to ask the Kerberos administrator to send you a copy of the file.

To connect to an Informatica domain that uses Kerberos authentication, perform the following tasks:

### **Copy the Kerberos configuration file to the client machines.**

Copy the Kerberos configuration file `krb5.com` to a directory on the client machine that is accessible to the client application.

### **Set the KRB5\_CONFIG environment variables with the Kerberos configuration file.**

Use the KRB5\_CONFIG environment variable to store the path and file name of the Kerberos configuration file, `krb5.conf`. You must set the KRB5\_CONFIG environment variable on each machine that hosts an Informatica client.

For more information about Kerberos authentication and the configuration file, see the *Informatica Security Guide*.

## Install the Drivers on Windows

Before you can connect to an SQL data service from a third-party client tool, you must install or upgrade the Informatica Data Services JDBC/ODBC drivers. Install the drivers on the machine that you connect to the SQL data service from. You can install the drivers on multiple machines in graphical mode or silent mode.

The Informatica JDBC/ODBC driver version must match the Informatica version.

## Install the Drivers in Graphical Mode

To install or upgrade the drivers in graphical mode, run the Informatica JDBC/ODBC driver installer.

1. Close all other applications.
2. Navigate to the root directory of the extracted installer files.
3. Run `install.bat`.
4. Choose to install or upgrade the drivers.
5. Click **Next**.

The **Installation Prerequisites** page displays the system requirements. Verify that all installation requirements are met before you continue the installation.

6. Click **Next**.
7. On the **Installation Directory** page, enter the absolute path for the installation directory.
8. On the **Pre-Installation Summary** page, review the installation information, and click **Install**.

The installer copies the driver files to the installation directory. The **Post-Installation Summary** page indicates whether the installation completed successfully.

9. Click **Done**.

You can view the installation log files to get more information about the tasks performed by the installer.

## Install the Drivers in Silent Mode

To install or upgrade the drivers without user interaction, install in silent mode. Use a properties file to specify the installation options. The driver installer reads the file to determine the installation options.

Use the silent mode installation to install the drivers on multiple machines on the network or to standardize the installation across machines.

1. Use a text editor to open and change the values of the properties in the file.

The following table describes the installation properties that you can change:

Property	Description
INSTALL_TYPE	Indicates whether to install or upgrade the drivers. <ul style="list-style-type: none"><li>• Enter 0 to install the drivers.</li><li>• Enter 1 to upgrade the drivers.</li></ul> Default is 0.
USER_INSTALL_DIR	Informatica driver installation directory. For example, C:\\Informatica\\<version>.

2. Save the properties file.
3. Navigate to the root directory of the extracted installer files.
4. To run the silent installation, double-click `silentInstall.bat`.

The silent installer runs in the background. The silent installation is complete when the `Informatica_<Version>_Driver_InstallLog.log` file is created in the root directory.

The silent installation fails if you incorrectly configure the properties file or if the installation directory is not accessible. View the installation log file in `SYSTEM_DRIVE_ROOT`. For example, C:\\`silentErrorLog.log`. Correct the errors, and then run the silent installation again.

## Install the Drivers on UNIX

Before you can connect to an SQL data service from a third-party client tool, you must install or upgrade the Informatica Data Services JDBC/ODBC drivers. Install the drivers on the machine that you want to connect to the SQL data service from. You can install the drivers on multiple machines in console mode or silent mode.

Before you can connect to an SQL data service on UNIX, you must perform the following tasks:

1. Install or upgrade the Informatica JDBC/ODBC drivers.
2. Configure the shared library environment variable.
3. Configure the ODBC files.

## Install the Drivers in Console Mode

To install or upgrade the drivers in console mode, run the Informatica JDBC/ODBC driver installer.

1. Log in to the machine with a system user account.
2. Close all other applications.
3. Navigate to the root directory of the extracted installer files.



4. At the command prompt, extract the install media.
5. Run `install.sh`.
6. Select one of the following options:

Option	Description
1	Install the drivers.
2	Upgrade the drivers.

7. Press **Enter**.  
The Installation Prerequisites section displays the system requirements.
8. Verify that all installation requirements are met before you continue the installation.
9. Press **Enter**.  
The Installation Directory section appears.
10. Enter the absolute path for the installation directory.
11. Press **Enter**.
12. In the Pre-Installation Summary section, review the installation information, and then press **Enter**.
13. Press **Enter**.

For more information about the install tasks, see the installation debug log.

## Install the Drivers in Silent Mode

To install or upgrade the drivers without user interaction, install in silent mode. Use a properties file to specify the installation options. The driver installer reads the file to determine the installation options.

Use the silent mode installation to install the drivers on multiple machines on the network or to standardize the installation across machines.

1. Use a text editor to open and change the values of the properties in the file.

The following table describes the installation properties that you can change:

Property	Description
<code>INSTALL_TYPE</code>	Indicates whether to install or upgrade the drivers. <ul style="list-style-type: none"><li>• Enter 0 to install the drivers.</li><li>• Enter 1 to upgrade the drivers.</li></ul> Default is 0.
<code>USER_INSTALL_DIR</code>	Informatica driver installation directory. For example, <code>C:\\Informatica\\&lt;version&gt;</code> .

2. Save the properties file.
3. Navigate to the root directory of the extracted installer files.

4. To run the silent installation, double-click `silentInstall.bat`.

The silent installer runs in the background. The silent installation is complete when the `Informatica_<Version>_Driver_InstallLog.log` file is created in the root directory.

The silent installation fails if you incorrectly configure the properties file or if the installation directory is not accessible. View the installation log file in `SYSTEM_DRIVE_ROOT`. For example, `C:\silentErrorLog.log`. Correct the errors, and then run the silent installation again.

## JDBC Connections

You can connect to an SQL data service through a JDBC client tool such as the SQL Squirrel client.

To connect to an SQL data service through a JDBC client tool, you must configure the JDBC connection.

### JDBC Connection Properties

When you configure the JDBC connection, you configure the connection properties in the driver. Optionally, you can configure the Data Integration Service parameters for the connection.

The following table describes the JDBC options and values that you enter:

Option	Value
Class name	<code>com.informatica.ds.sql.jdbcdrv.INFADriver</code>
JDBC URL	<p>Connection string for the JDBC connection.</p> <p>Use the following JDBC connection string for the different types of authentication:</p> <ul style="list-style-type: none"><li>- Native or LDAP Authentication: <code>jdbc:informatica:sqls/&lt;optional security domain&gt;&lt;optional user name&gt;/&lt;optional user password&gt;@&lt;domain host name&gt;:&lt;domain HTTP port&gt;?dis=&lt;Data Integration Service name&gt;&amp;sqls=&lt;run-time SQL data service name&gt;&amp;authType=native_uid</code></li><li>- Kerberos with keytab: <code>jdbc:informatica:sqls/&lt;optional security domain&gt;&lt;SPN&gt;@&lt;domain host name&gt;:&lt;domain HTTP port&gt;?dis=&lt;Data Integration Service name&gt;&amp;sqls=&lt;run-time SQL data service name&gt;&amp;keyTabFileLocation=&lt;keytab-location&gt;&amp;authType=kerberos_keytab</code></li><li>- Kerberos with user name and password: <code>jdbc:informatica:sqls/&lt;optional security domain&gt;&lt;user name&gt;/&lt;user password&gt;@&lt;domain host name&gt;:&lt;domain HTTP port&gt;?dis=&lt;Data Integration Service name&gt;&amp;sqls=&lt;run-time SQL data service name&gt;&amp;authType=kerberos_uid</code></li><li>- Logged in user: <code>jdbc:informatica:sqls/@&lt;domain host name&gt;:&lt;domain HTTP port&gt;?dis=&lt;Data Integration Service name&gt;&amp;sqls=&lt;run-time SQL data service name&gt;&amp;authType=sso</code></li></ul>

The following table describes the Data Integration Service parameters that you can configure:

Parameter	Value
application	ACCESS Enter this parameter when you query Microsoft Access virtual tables that contain date columns. When you configure the ODBC driver with this parameter, the Data Integration Service converts Microsoft Access date data to the date/time data type. The parameter applies only to Microsoft Access date data.
optimizeLevel	Sets the mapping optimization level. Enter one of the following values: <ul style="list-style-type: none"><li>- 0. Sets the optimization level to None.</li><li>- 1. Sets the optimization level to Minimal.</li><li>- 2. Sets the optimization level to Normal.</li><li>- 3. Sets the optimization level to Full.</li></ul> The default value is 1.
highPrecision	Runs mappings in high-precision mode. Enter <code>true</code> or <code>false</code> .
defaultDateFormat defaultTimeFormat defaultTimeStampFormat	Specifies the date and time formats. Enter one of the following values: <ul style="list-style-type: none"><li>- YYYY-MM-DD HH24:MI:SS</li><li>- YYYY/MM/DD HH24:MI:SS</li><li>- YYYY/MM/DD</li><li>- MM/DD/YYYY</li><li>- MM/DD/YYYY HH24:MI:SS</li><li>- DD/MM/YY</li><li>- DD.MM.YY</li><li>- DD-MON-YY</li><li>- DD/MM/YY HH24:MI:SS</li><li>- DD.MM.YY HH24:MI:SS</li></ul>
dumpMapping	Creates XML files for SQL query mappings and stores them in the following location: <code>&lt;Informatica installation directory&gt;\tomcat\bin\dslogs\sql</code> . If a query fails, you can send these files to Informatica Global Customer Support for analysis. Enter <code>true</code> or <code>false</code> . The default value is <code>false</code> .
ResultSetCacheExpirationPeriod	Amount of time in milliseconds that a result set is available for use after it is populated. For example, if the value is 0, result set caching is disabled. If the value is 5, the result set is available for 5 milliseconds after it is populated.

## Configure JDBC Connections on Windows

To connect to an SQL data service through a JDBC client tool, configure the JDBC client tool.

1. Open the JDBC client tool.
2. Enter the Class name and JDBC URL.
3. Optionally, configure the Data Integration Service parameters.

To configure the Data Integration Service parameters, append the parameters and values to the JDBC connection string.

Use the following syntax to configure the optional parameters:

```
<name1>=<value1>&<name2>=<value2>
```

For example,

```
optimizeLevel=0&highPrecision=false
```

## Configure JDBC Connections on UNIX

Before you can connect to an SQL data service on UNIX, you must configure the CLASSPATH environment variable and configure the client tool.

### Configure the CLASSPATH

To connect to an SQL data service through a JDBC client tool, configure the CLASSPATH environment variable to include the JDBC driver. The Informatica JDBC driver is compiled with Java 7.

The Informatica JDBC driver is installed in the following location: `<Informatica installation directory>\tools\jdbcdrv\infadsjdbc.jar`.

### Configure the JDBC Client Tool

To connect to an SQL data service through a JDBC client tool, configure the JDBC client tool.

1. Open the JDBC client tool.
2. Enter the Class name and JDBC URL.
3. Optionally, configure the Data Integration Service parameters.

To configure the Data Integration Service parameters, append the parameters and values to the JDBC connection string.

Use the following syntax to configure the optional parameters:

```
<name1>=<value1>&<name2>=<value2>
```

For example,

```
optimizeLevel=0&highPrecision=false
```

## Troubleshooting JDBC Connections

**I am connecting to an SQL data service through a JDBC client tool. I get an error when I try to bind data to a bigint column through a parameterized query.**

The Informatica JDBC driver does not support the `setBigInt` method. If you use `setBigInt` in a parameterized query, the Java client code fails at compilation.

If you need to bind data to a bigint column, use the `setLong` method instead of the `setBigInt` method. For example, the following code does not cause a JDBC driver error:

```
String sql = "select * from VT where COL_BIGINT = ?";
PreparedStatement stmt = connection.prepareStatement(sql);
stmt.setLong (1,new Long("9223372036854775807"));
```

**A third-party library included in `infadsjdbc.jar` causes a conflict on my machine.**

The `infadsjdbc.jar` file contains the Informatica JDBC driver and the following third-party libraries required by the JDBC driver:

- commons-codec-1.3.jar
- commons-httpclient-3.1.jar
- commons-logging-1.1.jar
- commons-pool-1.4.jar
- FastInfoset-1.2.3.jar

- log4j-1.2.12.jar
- spring-2.5.jar

If the JDBC client machine includes an instance of one of the bundled third-party libraries, a conflict might occur.

To resolve third-party library conflicts, use the `infadsjdbcclight.jar` file that is also installed by the Informatica JDBC/ODBC driver installation program. `infadsjdbcclight.jar` contains the Informatica JDBC driver and is installed in the following location: `<Informatica installation directory>\jdbcdrv\infadsjdbc`. The `infadsjdbc` folder also contains all of the third-party libraries that are included with `infadsjdbc.jar`.

To use `infadsjdbcclight.jar`, modify the CLASSPATH environment variable with the location of `infadsjdbcclight.jar` and with the location of the third-party libraries that do not cause a conflict. For example, if the JDBC client machine includes an instance of the `spring-2.5.jar` library, remove `infadsjdbc.jar` from the CLASSPATH environment variable. Then, add the following files to the CLASSPATH:

```
<Informatica installation directory>\jdbcdrv\infadsjdbc\infadsjdbcclight.jar
<Informatica installation directory>\jdbcdrv\infadsjdbc\commons-codec-1.3.jar
<Informatica installation directory>\jdbcdrv\infadsjdbc\commons-httpclient-3.1.jar
<Informatica installation directory>\jdbcdrv\infadsjdbc\commons-logging-1.1.jar
<Informatica installation directory>\jdbcdrv\infadsjdbc\commons-pool-1.4.jar
<Informatica installation directory>\jdbcdrv\infadsjdbc\FastInfoset-1.2.3.jar
<Informatica installation directory>\jdbcdrv\infadsjdbc\log4j-1.2.12.jar
```

## ODBC Connections

You can connect to an SQL data service through an ODBC client tool such as IBM Cognos.

To connect to an SQL data service through a JDBC client tool, you must configure the ODBC connection.

### ODBC Connection Properties

When you configure the ODBC connection, you configure the connection properties in the driver. Optionally, you can configure the Data Integration Service parameters for the connection.

The following table describes the properties that you configure for the driver:

Option	Definition
Authentication Mode	The Authentication Mode parameter can have one of the following values: <ul style="list-style-type: none"> <li>• Native or LDAP authentication.</li> <li>• Kerberos with keytab file.</li> <li>• Kerberos with username &amp; password.</li> <li>• Logged in user.</li> </ul>
DSN Name	Any valid data source name.

Option	Definition
Host Name	Machine that hosts the Informatica domain.
Port	Informatica domain HTTP port number.
Data Integration Service name	Data Integration Service that runs the application that contains the SQL data service.
SQL data service name	Name of the SQL data service that contains the virtual tables you want to query or the virtual stored procedures that you want to run. The run-time SQL data service name includes the application name that contains the SQL data service and uses the following format: <application name>.<SQL data service name>
User Name	Informatica domain user name. Required if you select the <b>Native or LDAP Authentication</b> or <b>Kerberos with username &amp; password</b> authentication mode.
Password	Informatica domain user password. Required if you select the <b>Native or LDAP Authentication</b> or <b>Kerberos with username &amp; password</b> authentication mode.
Security Domain	Security domain for the Informatica domain user account. Required if the user account is in an LDAP security domain.
Keytab for SQL Data Service	Absolute path and file name for the keytab file on the client machine. Required if you select the <b>Kerberos with keytab file</b> authentication mode.
SPN for SQL Data Service	Service principal name for the user account. Required if you select the <b>Kerberos with keytab file</b> authentication mode.

The following table describes the Data Integration Service parameters that you can configure:

Parameter	Value
application	ACCESS Enter this parameter when you query Microsoft Access virtual tables that contain date columns. When you configure the ODBC driver with this parameter, the Data Integration Service converts Microsoft Access date data to the date/time data type. The parameter applies only to Microsoft Access date data.
optimizeLevel	Sets the mapping optimization level. Enter one of the following values: <ul style="list-style-type: none"> <li>- 0. Sets the optimization level to None.</li> <li>- 1. Sets the optimization level to Minimal.</li> <li>- 2. Sets the optimization level to Normal.</li> <li>- 3. Sets the optimization level to Full.</li> </ul> The default value is 1.
highPrecision	Runs mappings in high-precision mode. Enter <code>true</code> or <code>false</code> .

Parameter	Value
defaultDateFormat defaultTimeFormat defaultTimeStampFormat	Specifies the date and time formats. Enter one of the following values: <ul style="list-style-type: none"> <li>- YYYY-MM-DD HH24:MI:SS</li> <li>- YYYY/MM/DD HH24:MI:SS</li> <li>- YYYY/MM/DD</li> <li>- MM/DD/YYYY</li> <li>- MM/DD/YYYY HH24:MI:SS</li> <li>- DD/MM/YY</li> <li>- DD.MM.YY</li> <li>- DD-MON-YY</li> <li>- DD/MM/YY HH24:MI:SS</li> <li>- DD.MM.YY HH24:MI:SS</li> </ul>
dumpMapping	Creates XML files for SQL query mappings and stores them in the following location: <Informatica installation directory>\tomcat\bin\dslogs\sql. If a query fails, you can send these files to Informatica Global Customer Support for analysis. Enter true or false. The default value is false.
ResultSetCacheExpirationPeriod	Amount of time in milliseconds that a result set is available for use after it is populated. For example, if the value is 0, result set caching is disabled. If the value is 5, the result set is available for 5 milliseconds after it is populated.

## Configure ODBC Connections on Windows

Before you connect to an SQL data service through an ODBC client tool on Windows, install the ODBC driver on the client machine using the Informatica Data Services JDBC/ODBC driver installer. Install a 32-bit or 64-bit ODBC driver depending on the version of the third-party client tool you are using. After you install the driver, configure the Data Source Name (DSN).

- Select the correct version of **ODBC Data Source Administrator**.  
A 32-bit version of the Microsoft Windows operating system includes the following version of the Microsoft ODBC Data Source Administrator tool (Odbcad32.exe):
  - The 32-bit version of the Odbcad32.exe file is located in the %systemdrive%\Windows\System32 folder.
A 64-bit version of the Microsoft Windows operating system includes the following versions of the Microsoft ODBC Data Source Administrator tool (Odbcad32.exe):
  - The 32-bit version of the Odbcad32.exe file is located in the %systemdrive%\Windows\SysWoW64 folder.
  - The 64-bit version of the Odbcad32.exe file is located in the %systemdrive%\Windows\System32 folder.
- In the ODBC Data Source Administrator, click **Add**.
- Select **Informatica Data Services ODBC Driver <version>**.
- Click **Finish**.  
The **Create a New Data Service** window appears.
- Configure the authentication mode.
- Configure the driver properties.
- Optionally, configure the Data Integration Service parameters.

Use the following syntax to configure the optional parameters:

```
<name1>=<value1>&<name2>=<value2>
```

For example,

```
optimizeLevel=0&highPrecision=false
```

8. Click **Test Connection** to verify that the connection is valid and then click **OK**.

## Configure ODBC Connections on UNIX

Before you can connect to an SQL data service on UNIX, you must configure the shared library environment variable and configure the `odbc.ini` and the `odbcinst.ini` files.

### Configure the Shared Library Environment Variable

Configure the shared library environment variable to include the driver libraries and directory where the driver manager library files reside.

Configure the shared library environment variable based on the operating system. The following table lists the shared library environment variable for each operating system:

Operating System	Shared Library Environment Variable
AIX	LIBPATH
HP-UX	SHLIB_PATH
Solaris	LD_LIBRARY_PATH

Configure the shared library environment variable to include the following directories:

- The directory where the ODBC driver libraries reside. The driver libraries reside in `libinfadsodbc`. `libinfadsodbc` is found in `<Informatica installation directory>/tools/odbcdrv`.
- The directory where the driver manager library files reside. Use the `unixODBC` driver manager. For more information about the location of the `unixODBC` driver manager, contact your system administrator.

### Configure the ODBC Files

Before you can connect to an SQL data service on UNIX, you must configure the `odbc.ini` and `odbcinst.ini` files.

Store the `odbcinst.ini` file in the `<DriverHome>` directory.

1. Edit the `odbc.ini` file or copy the `odbc.ini` file to the root directory and edit it.

This file exists in the `$ODBCHOME` directory.

```
$ cp $ODBCHOME/odbc.ini $HOME/.odbc.ini
```

2. Add an entry for the ODBC user under the section `[<DSN>]`.

For example:

```
[<DSN>]
DataIntegrationService=<Data Integration Service name>
SQLDataService=<runtime SQL data service name>
Driver=$ODBC_DRIVER_INSTALL_LOCATION/bin/$OS/libinfadsodbc.so
HostName=<domain host name>
Port=<domain HTTP port>
```



```
Authentication Mode=<type>
Optional Parameters=defaultDateFormat=DD/MM/YYYY&defaultTimeStampFormat=DD/MM/YYYY
HH24:MI:SS
WCHARLengthInChars=true
```

**Note:** Configure WCHARLengthInChars for MicroStrategy.

3. Add the driver and setup entries to the `odbcinst.ini` file.

```
[Informatica Data Services ODBC Driver <version>]
Driver=<Complete path to driver>
Setup=<Complete path to driver>
```

For example:

```
Driver=$ODBC_DRIVER_INSTALL_LOCATION/odbcdrv/libinfadsodbc.so
Setup=$ODBC_DRIVER_INSTALL_LOCATION/odbcdrv/libinfadsodbc.so
```

## CHAPTER 5

# Third-Party Client Tool Configuration

This chapter includes the following topics:

- [Third-Party Client Tool Configuration Overview, 58](#)
- [BusinessObjects, 59](#)
- [IBM Cognos Configuration, 60](#)
- [MicroStrategy Configuration, 65](#)
- [Oracle Business Intelligence Enterprise Edition 11g Configuration, 71](#)
- [Oracle Database Gateway Configuration, 72](#)
- [QlikView Configuration, 72](#)
- [SQL Server Business Intelligence Development Studio Configuration, 74](#)
- [Squirrel SQL Client Configuration, 75](#)
- [Tableau Configuration, 78](#)
- [Toad for Data Analysts Configuration, 78](#)
- [WinSQL Configuration, 79](#)
- [Troubleshooting Third-Party Client Tools, 80](#)

## Third-Party Client Tool Configuration Overview

After you install and configure the JDBC or ODBC drivers, you can connect to the third-party client tool to access an SQL data service.

You can connect to an SQL data service through the following third-party client tools:

- BusinessObjects
- IBM Cognos
- MicroStrategy
- Oracle Business Intelligence Enterprise Edition 11g
- Oracle Database Gateway
- QlikView
- Squirrel SQL Client

- Tableau
- Toad for Data Analysts
- WinSQL

If you are using a 32-bit third-party client tool, install a 32-bit ODBC driver. If you are using a 64-bit third-party client tool, install a 64-bit ODBC driver. For more information on configuring drivers, see [“Configure ODBC Connections on Windows” on page 55](#).

## BusinessObjects

You can access the virtual data in an SQL data service through SAP BusinessObjects. Use the Information Design Tool provided by SAP BusinessObjects to extract, define, and manipulate metadata for BusinessObject BI applications.

Use the Information Design Tool to create a project, to define data source connections, and to import metadata. Create the data foundation with required connections and then set up a business layer. When you have the required metadata in the form of universe, you can publish the universe to the BusinessObjects Server. SAP BusinessObjects uses universes created by the Information Design Tool for data analysis and to query the data and generate enterprise reports.

## BusinessObjects Configuration

You can access the virtual data in an SQL data service through a BusinessObjects universe. Import metadata from the SQL data service into the universe. Use a BusinessObjects application to query the data or generate reports.

To configure BusinessObjects to access an SQL data service, complete the following tasks:

1. Start the Information Design Tool.
2. Click **File > New > Project**.
3. Provide the project name and location and click **Finish**.
4. Under **Local Projects**, right-click on the project name and select **New > Relational Connection**.  
The **New Relational Connection** dialog box appears.
5. Provide the resource name and a description and click **Next**.
6. In the Hierarchical List, expand **Generic** and choose **Generic ODBC Datasource > Select ODBC Drivers**.
7. Click **Next**.  
The **Login Parameters** page appears.

8. Enter the connection information.

The following table describes the connection options:

Option	Description
Authentication Mode	Select <b>Use specified username and password</b> .
User name	Informatica domain user name.
Password	Informatica domain user password.
Data source name	ODBC data source name for the SQL data service.

9. Click **Test Connection** to test the connection.

The BusinessObjects application displays a **Test Successful** message.

10. Click **Finish**.

After you perform this task, you can import metadata from an SQL data service into the universe and generate reports based on the data.

## IBM Cognos Configuration

IBM Cognos Framework Manager is a modeling tool that allows you to create and manage business-related metadata for use in all Cognos BI applications. Use Framework Manager to create a project. Within a project, you can create data source connections and import data source metadata.

When you define a data source in Framework Manager, you create an ODBC connection to the SQL data service. Cognos uses the information in the connection to connect to the SQL data service. After you create the ODBC connection and import the SQL data service metadata, you can create a Cognos model based on the SQL data service. Use the model to run queries and generate reports in Cognos BI applications.

To run queries that include aggregations, the Cognos configuration file must include Informatica ODBC driver information.

To configure IBM Cognos to access an SQL data service, complete the following tasks:

1. Update the IBM Cognos configuration file to include the Informatica ODBC driver information.
2. Create an ODBC connection to the SQL data service, and import SQL data service metadata in a Cognos project.

## Step 1. Update the IBM Cognos Configuration File

Update the IBM Cognos configuration file to include Informatica ODBC driver information. The ODBC driver information in the configuration file enables the Data Integration Service to handle SQL queries that include aggregations. If you do not update the configuration file and you query an SQL data service from an IBM Cognos application, the query might fail if it includes aggregations.

The IBM Cognos configuration file, `cogdmod.ini`, is in the following directories:

- C:\Program Files\cognos\tools\c8\bin\
- C:\Program Files\cognos\c8\bin\

You must update both copies of the configuration file.

1. Stop the Cognos service.
2. Back up both copies of `cogdmod.ini`.
3. Add the following entry to both copies of `cogdmod.ini`:

```
[Exceptions Set Operators DRIVER:INFADSODBC.DLL]
Distinct="distinct"
Except="except"
Except_All=T
Intersect="intersect"
Intersect_All=T
Union="union"
Union_All=T
[Exceptions Tables DRIVER:INFADSODBC.DLL]
Derived=T
Joined=T
Simple=T
UniqueName_Prefix="coguda"
[Exceptions OLAP Functions DRIVER:INFADSODBC.DLL]
Olap_Max="max"
Olap_Min="min"
Olap_Sum="sum"
Olap_Avg="avg"
Olap_Count="count"
Olap_Count_Star="count(*)"
Olap_Distinct="distinct"
```

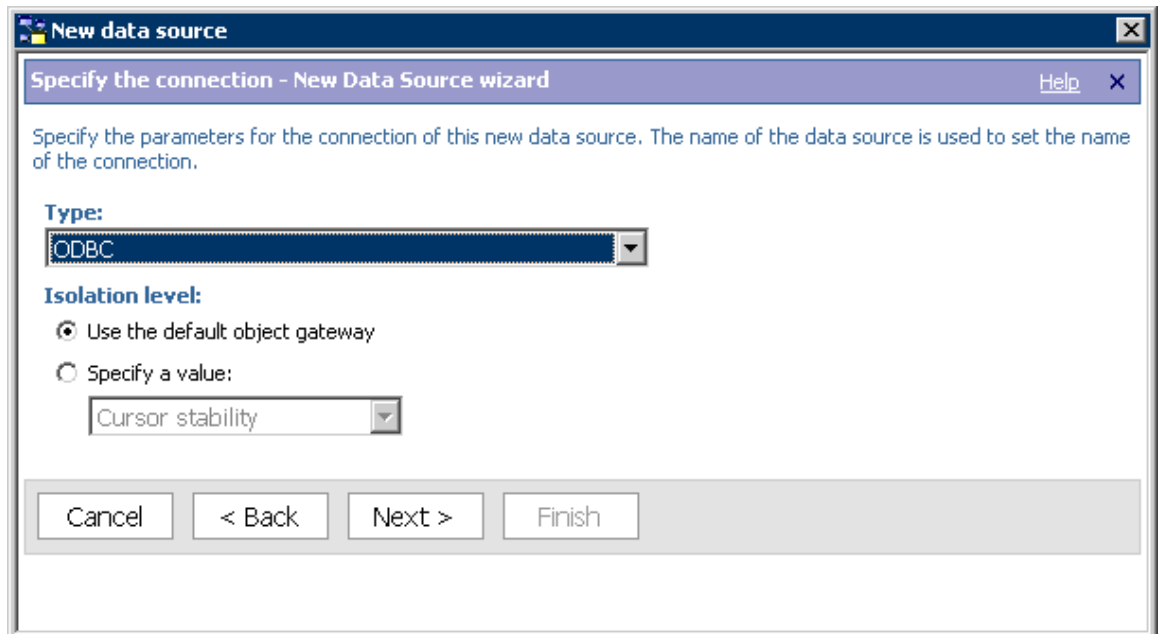
4. Restart the Cognos service.

## Step 2. Create the ODBC Connection and Import Metadata

Create the ODBC connection and import SQL data service metadata into a project. Use Cognos Framework Manager to create the project. Run the Framework Manager **Run Metadata** wizard to create the connection and import metadata.

1. Start Cognos Framework Manager.
2. Create a project.
3. Start the **Run Metadata** wizard.

4. In the **Select Metadata Source** window, click **Data Sources**.
5. Click **Next**.  
The **Select Data Source** window appears.
6. Click **New**.  
The **New Data Source** wizard appears.
7. In the name and description page, enter a name and optional description for the data source.
8. Click **Next**.
9. In the connection page, select the ODBC database type, select an isolation level, and click **Next**.



10. In the connection string page, enter the SQL data service ODBC data source name in the **ODBC data source** and **ODBC connect string** fields. Enter timeouts or sign-on information, if required. Enter the user ID and password if they are not part of the Informatica ODBC driver connect string.

**New data source**

**ODBC data source:**  
HypoStores\_Customer

**ODBC connect string:**  
HypoStores\_Customer

**Collation sequence:**

☐ Open asynchronously

☐ Unicode ODBC

**Timeouts**  
Specify the time in seconds, in which you want the database to connect or wait for your reply before timing out.

**Connect time:**  
0

**Reply time:**  
0

**Signon**  
Select whether a user ID and password is required in the connection string and, if so, whether to create a signon.

☐ User ID

☒ Password

☒ Create a signon that the Everyone group can use:

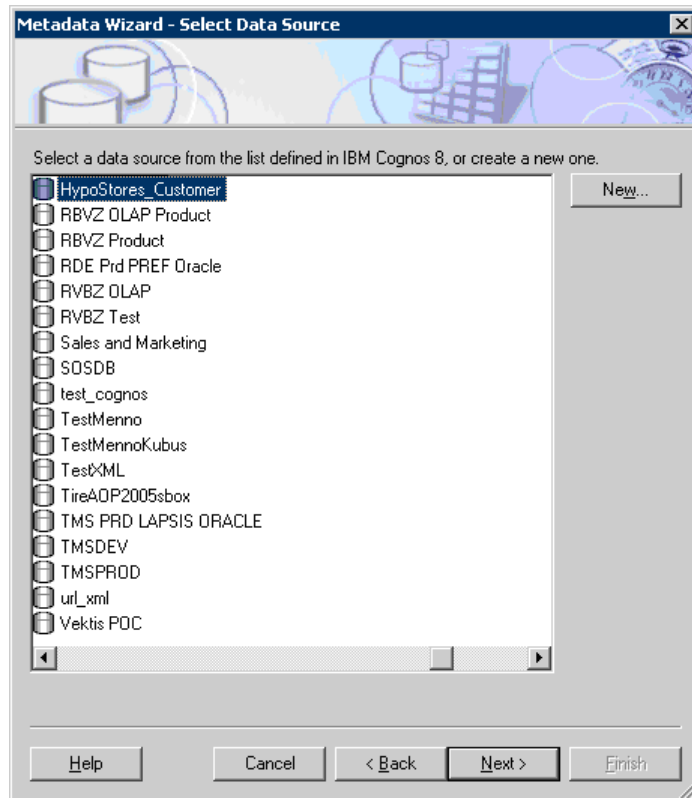
**User ID:**

**Password:**

**Confirm password:**

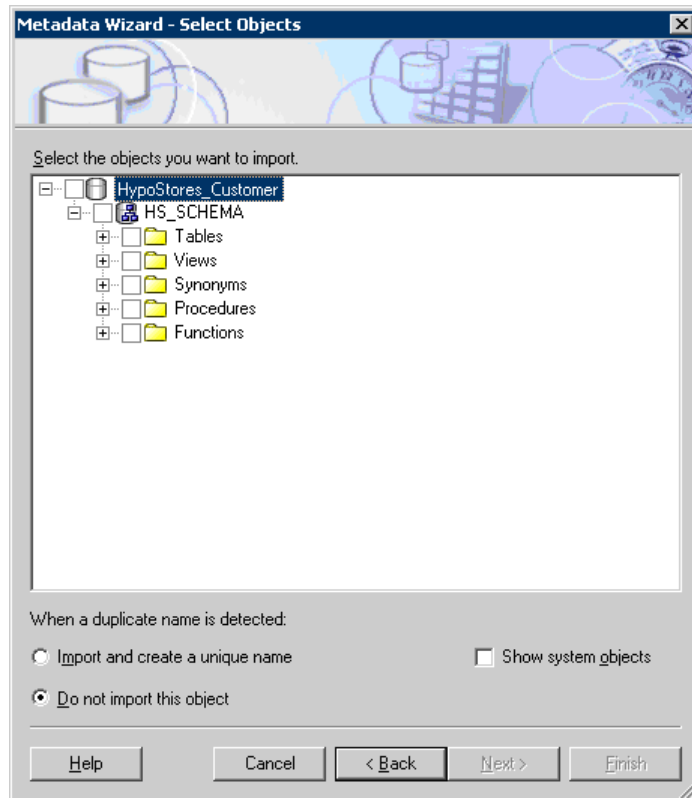
**Testing**  
[Test the connection...](#)

11. Click **Test the connection** to test the connection to the Informatica ODBC driver.
12. In the **Run Metadata** wizard, select the data source.



13. Click **Next**.
14. In the **Select Objects** page, select the objects you want to import and specify how the import handles duplicate object names.





15. Specify the criteria to use to create relationships and click Import.  
Framework Manager displays the number of imported objects and a list of objects that it could not import.
16. Click **Finish**.

## MicroStrategy Configuration

MicroStrategy is a business intelligence platform that allows you to analyze, distribute, and customize business information. MicroStrategy Desktop allows you to create projects and reports. Within a project, you can create data source connections and import data source metadata.

To configure MicroStrategy to access an SQL data service, complete the following tasks:

1. Create the database instance and connection.
2. Configure the SQL generation options.

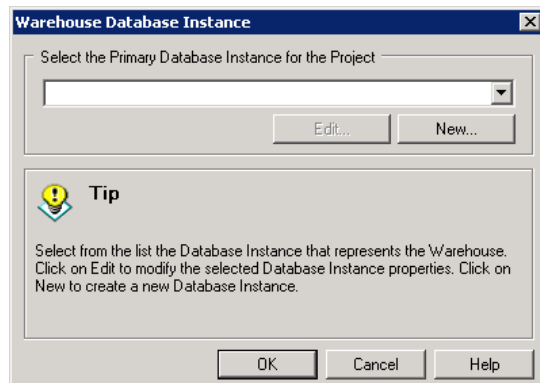
### Step 1. Create the Database Instance and Connection

Use MicroStrategy Desktop to create the database instance and database connection. MicroStrategy retrieves SQL data service metadata through the database instance.

1. Open MicroStrategy Desktop.
2. Create a project.

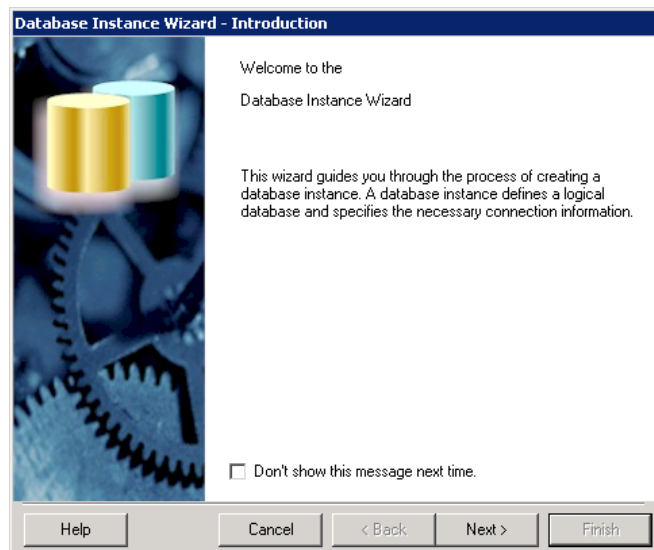
3. Select **Schema > Warehouse Catalog** to open the project Warehouse Catalog.

The **Warehouse Database Instance** dialog box appears.



4. Click **New**.

The **Database Instance Wizard** opens.



5. Click **Next**.
6. In the **Database Instance General Information** page, enter a name for the database instance and select **Generic DBMS** as the database type.

**Database Instance Wizard - Database Instance Definition**

Database Instance, General Information

Please enter the database instance information: database name (required), database description and database type (required).

Name:

Description:

Database type:

Help Cancel < Back Next > Finish

7. Click **Next**.
8. In the **ODBC Data Source Information** page, select the ODBC data source name for the SQL data service and enter the Informatica domain user name and password.

**Database Instance Wizard - Database Connection Definition**

ODBC Data Source Information

Please choose a DSN as the database connection and enter the information for database login.

Local system ODBC data sources:

Name
Excel_WH
FRAM_WH_AC
HRAM_WH_AC
HypoStores_Customer
JumpStart_Metadata
JumpStart_Warehouse

Configure ODBC

Database login information:

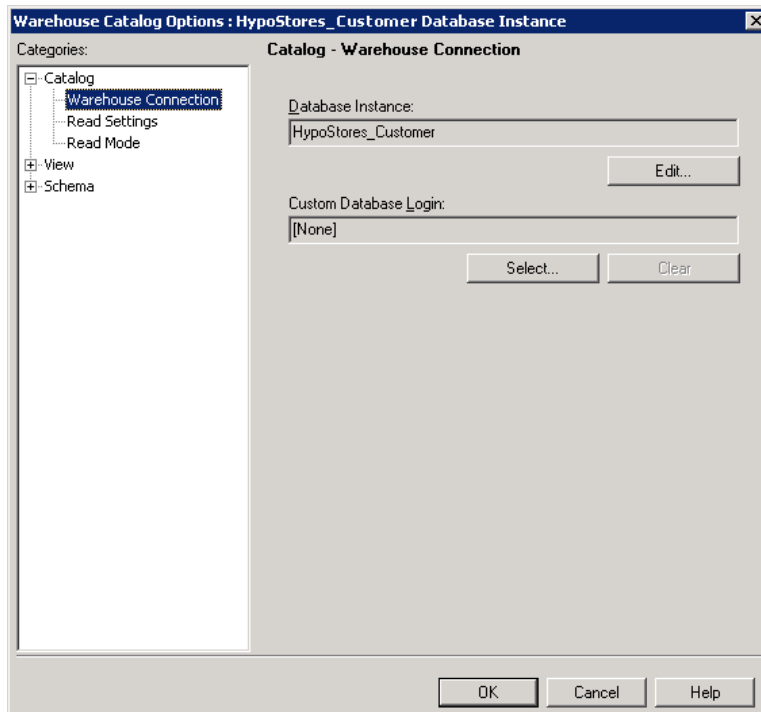
Database login:

Password:

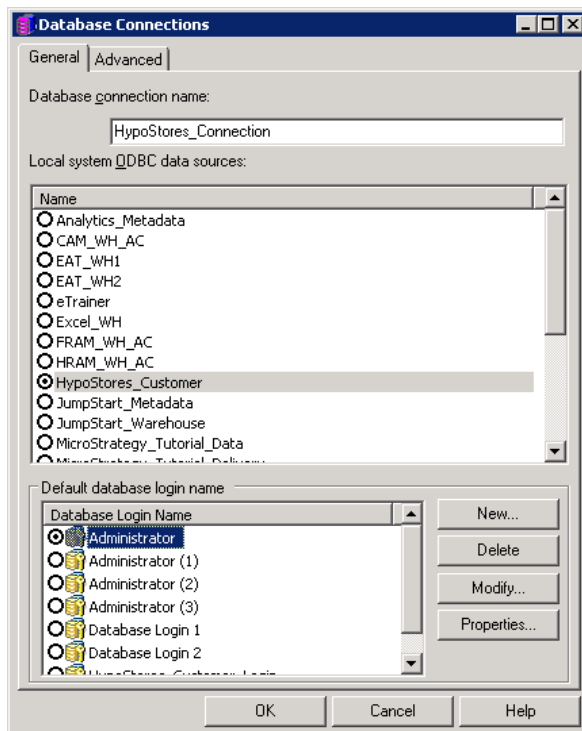
☐ Use network login id (Windows authentication)

Help Cancel < Back Next > Finish

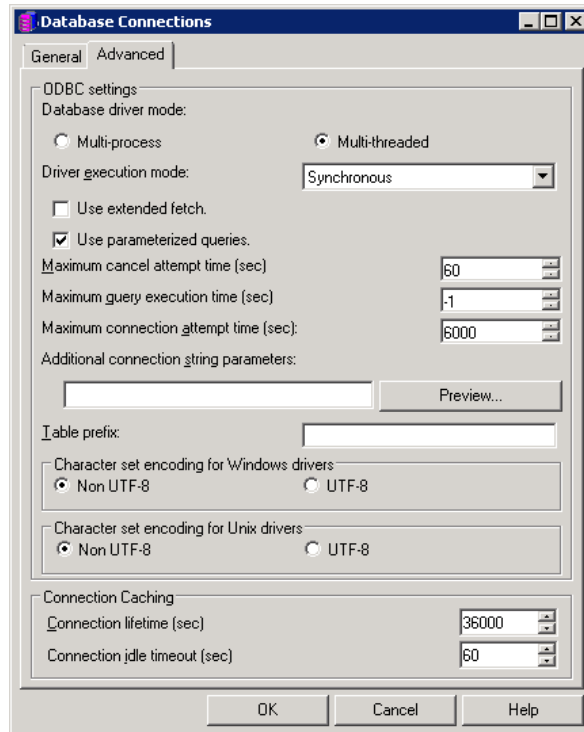
9. Click **Finish**.
10. Click **OK** to close the **Warehouse Database Instance** dialog box.
11. Select **Schema > Warehouse Catalog** to open the Warehouse Catalog.
12. Click **Options**.
13. Select **Warehouse Connection**.



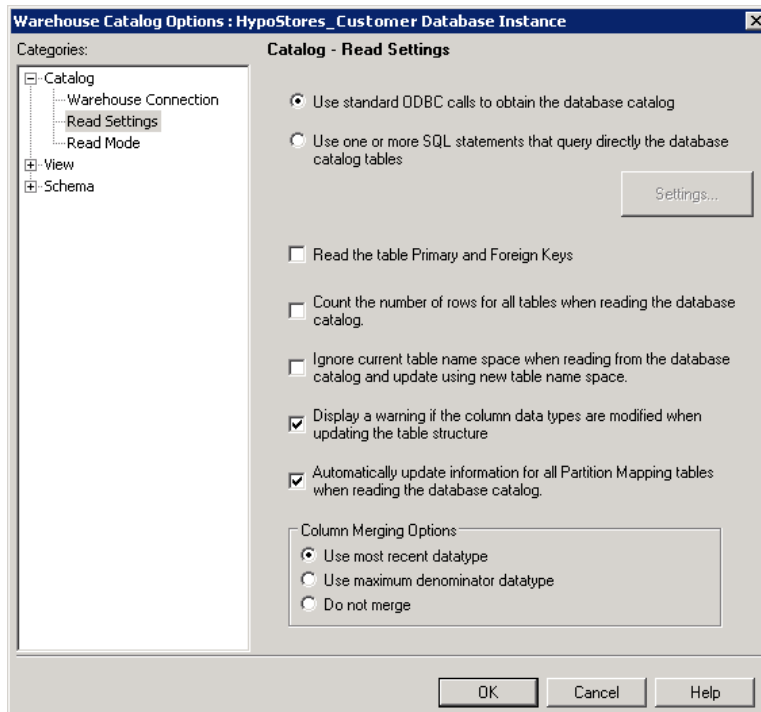
14. In the **Warehouse Connection** settings, select the database instance and click **Edit**.  
The **Database Instances** dialog box opens.
15. Click **New** to create a database connection.  
The **Database Connections** dialog box opens.



16. On the **General** tab, enter a database connection name and select the ODBC data source name for the SQL data service.
17. Click the **Advanced** tab.



18. Set the following options:
  - Select the **Multi-threaded** database driver mode.
  - Select the **Use parameterized queries** option.
  - Set the character set encoding option for Windows and UNIX drivers to **Non UTF-8**.
19. Click **OK** to close the **Database Connections** dialog box.
20. Click **OK** to close the **Database Instances** dialog box.
21. In the Warehouse Catalog, select **Read Settings**.

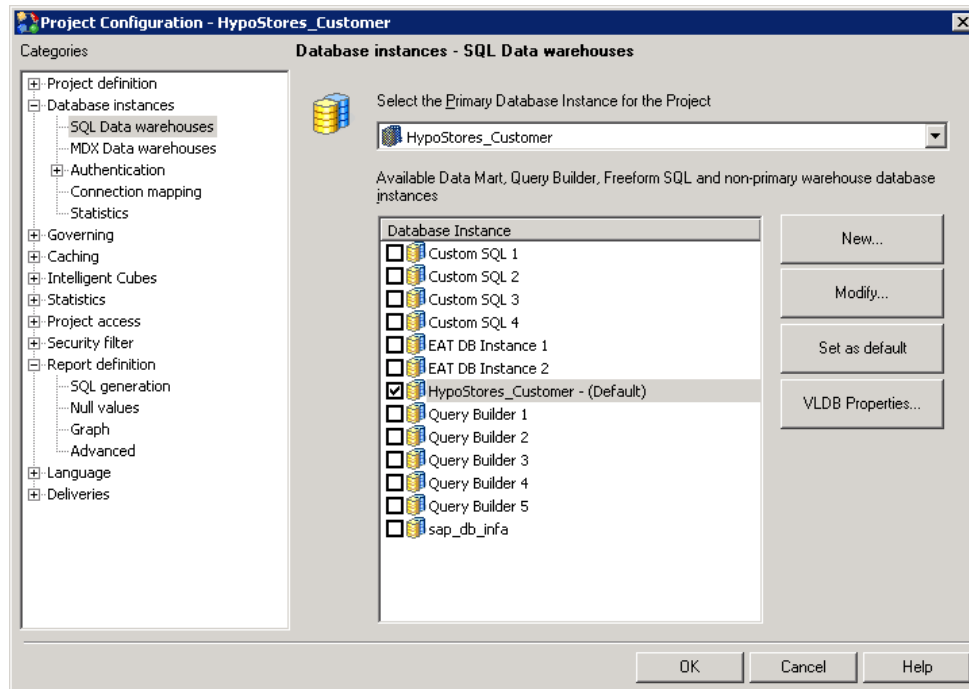


22. In the **Read Settings**, select **Use standard ODBC calls to obtain the database catalog**.
23. Click **OK**.
24. In the Warehouse Catalog, click **Save and Close** to save the changes.

## Step 2. Configure the SQL Generation Options

The virtual tables in an SQL data service are read-only tables. Configure the SQL generation options to prevent MicroStrategy Desktop users from trying to write data to the virtual database.

1. Select **Schema > SQL Generation Options** to open the SQL generation options.
2. In the **SQL Data Warehouses** settings, select the database instance you use to connect to the SQL data service.



3. Click **VLDB Properties** to edit the VLDB properties for the database instance.
4. Open the **Tables** settings.
5. In the **Drop Temp Tables Method** settings, set the drop temp table method to **Do nothing**.
6. In the **Intermediate Table Type** settings, set the intermediate table type to **Derived table**.
7. In the **Table Creation Type** settings, set the table creation type to **Implicit Table**.
8. In the **CREATE and INSERT Support** settings, select the **Create and insert are not supported option**.
9. Save and close the SQL generation options.

## Oracle Business Intelligence Enterprise Edition 11g Configuration

You can access the virtual data in an SQL data service through the Oracle Business Intelligence Enterprise Edition 11g (OBIEE 11g). Import metadata from an SQL data service into the OBIEE 11g to generate reports.

Import metadata from databases and other data sources to populate the physical layer. You can import schemas or portions of schemas from existing data sources. You can manually create objects in the physical layer.

To configure the OBIEE 11g to access an SQL data service, complete the following tasks:

1. Open the **Oracle BI Administration Tool**.
2. Create a new repository.  
The **New Repository** dialog opens with the **Repository** folder selected by default.
3. Name the file <RPDfilename>.rpd and enter the repository password twice.
4. Select the data source name created for the ODBC connection.

5. Select the **Metadata Types** to import and click **Next**.
6. Select the **Metadata Objects** and click **Finish**.

## Oracle Database Gateway Configuration

Oracle Database Gateway provides ODBC access to heterogeneous data sources through Oracle client software. You can use Oracle Database Gateway to query and join data from different data sources.

You can access the virtual data in an SQL data service through Oracle Database Gateway. You can use Oracle client software to query an SQL data service. The query returns data as if it came from an Oracle database. You can join virtual data from an SQL data service with data from other data sources.

You must change the configuration of Oracle Database Gateway to query an individual SQL data service. You can query multiple SQL data services, but you must configure individual ODBC connections for each SQL data service that you want to query.

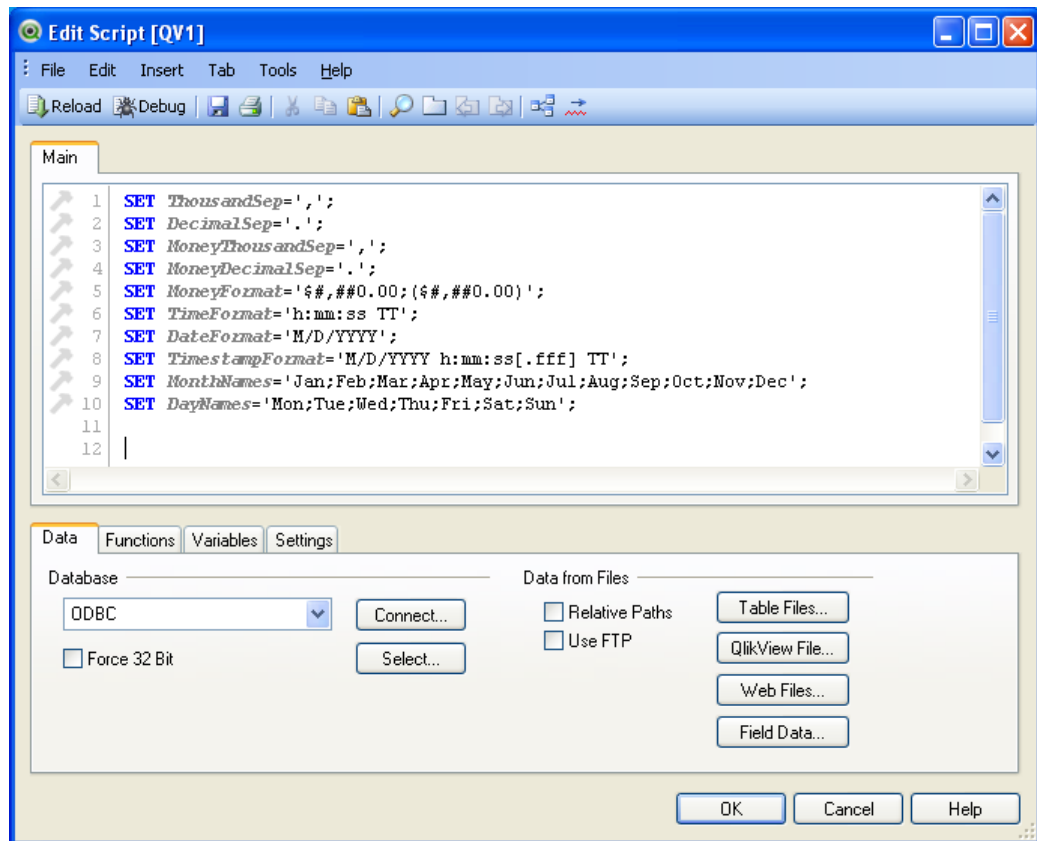
For more information about configuring Oracle Database Gateway with an SQL data service, see the Informatica How-To Library article, "Configuring Oracle Database Gateway for Informatica Data Services": <https://kb.informatica.com/h2l/HowTo%20Library/1/0715-OracleDBGatewaySQLDataServices-H2L.pdf>.

## QlikView Configuration

You can access the virtual data in an SQL data service through QlikView. To read data from an SQL data service into your QlikView document, use the Script Editor. The script that you create uses an ODBC connection to connect to and retrieve data from the SQL data service.

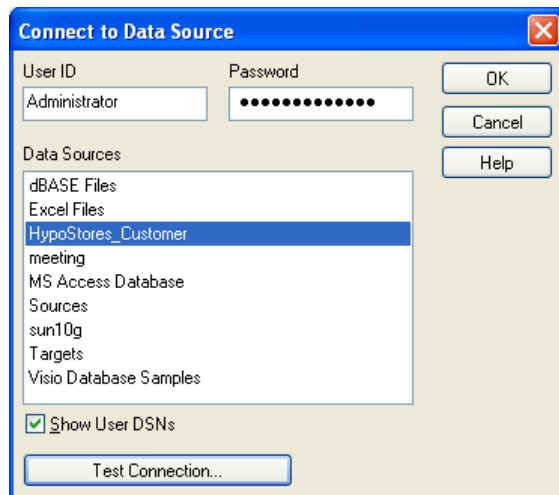
1. Create a QlikView document.
2. Select **File > Edit Script**.  
The **Edit Script** dialog box appears.





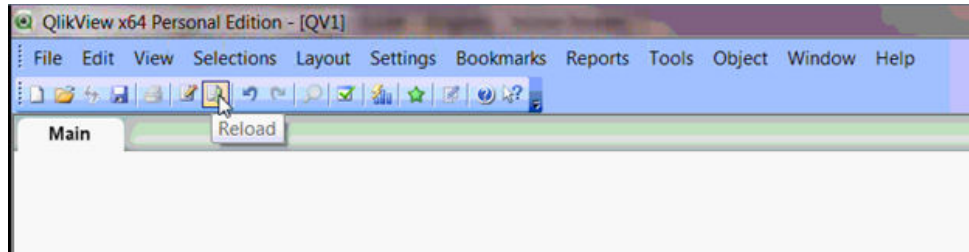
3. In the **Data** view, select ODBC as the database and click **Connect**.

The **Connect to Data Source** dialog box appears.



4. Select the ODBC data source name for the SQL data service and enter the user name and password for the Informatica domain user.
5. Click **Test Connection** to test the connection.
6. Click **OK** to close the connection results.
7. Click **OK** to close the **Connect to Data Source** dialog box.

8. In the **Data** view of the **Edit Script** dialog box, click **Select** to create an SQL SELECT statement that retrieves information from the SQL data service.
9. Click **OK**.
10. Run the script to retrieve data from the SQL data service.
  - a. Click the **Reload** button on the QlikView worksheet.



- b. Click **OK** and save the worksheet.
- c. After the script runs, click **Close**.
- d. Right-click within the worksheet and choose **Select Fields...**  
The **Sheet Properties** dialog box appears.
- e. Add the require fields to **Fields Displayed in Listboxes** and click **Apply > OK**. The listboxes appear in the QlikView worksheet.

## SQL Server Business Intelligence Development Studio Configuration

Business Intelligence Development Studio is an integrated development environment used for developing data analysis and business intelligence solutions that utilizes SQL Server Reporting Services. You can access the virtual data in an SQL data service with SQL Server Business Intelligence Development Studio.

Use the **Report Wizard** to connect to a data source and create a report. The Business Intelligence Development Studio imports data from an SQL data service based on the connection information.

To configure SQL Server Business Intelligence Development Studio, complete the following tasks:

1. Open the Business Intelligence Development Studio.
2. Create a new project and select **Business Intelligence Project** as the project type.
3. Select the **Report Server Project Wizard**.
4. Enter a project name and click **OK**.  
The **Report Wizard** dialog box appears.
5. Click **Next**.
6. Select **New Data Source**.
7. Enter the data source name and select **ODBC** as the type.
8. Click **Edit**.

The **Connection Properties** dialog box appears.

9. Enter the connection properties.

The following table describes the connection properties that you enter:

Option	Description
Data Source Name	Select the ODBC data source name for the SQL data service.
User ID	Informatica domain user name.
Password	Informatica domain user password.

10. Click **Test Connection** to validate the connection.
11. Click **Next**.
12. Click **Query Builder** and enter a username and password.  
The **Query Designer** window appears.
13. Select the **Add** table option.  
The **Add Table** dialog box appears.
14. Select tables and click **Add**.
15. Assign a relation or join tables manually.
16. Run an SQL query and verify that the data displays as expected.
17. Click **OK**.
18. Select **Tabular** or **Matrix** as the report type and click **Next**.
19. Select the available fields and click **Next**.
20. Enter a report name and click **Finish** to create the report.

## Squirrel SQL Client Configuration

You can access the virtual data in an SQL data service through Squirrel SQL Client. Squirrel SQL Client allows you to view the structure of a database and browse and query the data.

To configure Squirrel SQL Client to access an SQL data service, complete the following tasks:

1. Copy the Informatica JDBC driver to the Squirrel SQL Client library directory.
2. Create the Informatica JDBC driver and the database alias in Squirrel SQL Client.

After you perform these tasks, you can import data from an SQL data service into Squirrel SQL Client.

### Step 1. Copy the Driver File

Copy the Informatica JDBC driver to the Squirrel SQL Client library directory.

- Copy the Informatica JDBC driver, `infadsjdbc.jar`, from the following directory:

`<Informatica Installation Directory>\tools\jdbcdrv\`

To the following directory:

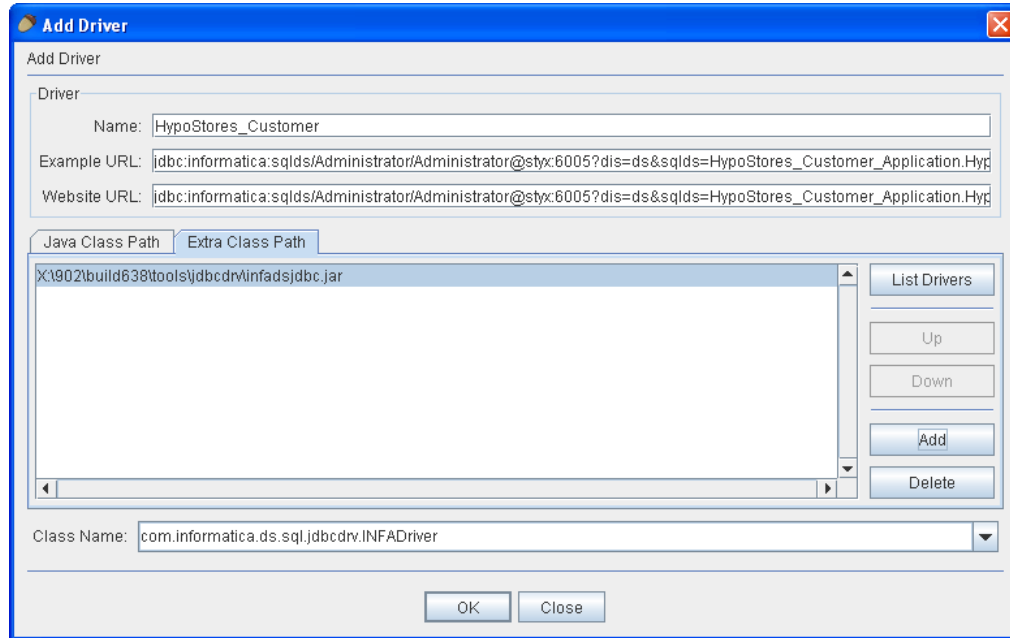
`<Squirrel SQL Client Installation Directory>\lib\`

## Step 2. Create the Driver and Alias

To read data from an SQL data service into Squirrel SQL Client, create a driver definition and a database alias. Squirrel SQL Client uses the driver definition and the database alias to enable you to view the SQL data service structure, browse the data in virtual tables, and enter SQL queries.

1. Open the **Drivers** list window.
2. Select **Drivers > New Driver**.

The **Add Driver** dialog box appears.



3. Enter the driver properties.

The following table describes the driver properties that you can enter:

Option	Description
Name	Informatica JDBC Driver
Example URL	jdbc:informatica:sqlDs/<optional security domain><optional user name>/<optional user password>@<domain host name>:<domain HTTP port>?dis=<Data Integration Service name>&sqlDs=<runtime SQL data servicename>
Website URL	jdbc:informatica:sqlDs/<optional security domain><optional user name>/<optional user password>@<domain host name>:<domain HTTP port>?dis=<Data Integration Service name>&sqlDs=<runtime SQL data servicename>
Extra Class Path	<Informatica Installation Directory>\tools\jdbcdrv\infadsjdbc.jar
Class Name	com.informatica.ds.sql.jdbcdrv.INFADriver

4. Click **OK**.

Squirrel SQL Client displays a message saying that driver registration is successful.

5. Open the **Aliases** list window.
6. Select **Aliases > New Alias**.

The **Add Alias** dialog box appears.

7. Enter the alias properties.

The following table describes the alias properties that you can enter:

Option	Description
Name	Alias name.
Driver	Select the new driver.
URL	jdbc:informatica:sqlsds/<optional security domain\><optional user name>/<optional user password>@<domain host name>:<domain HTTP port>?dis=<Data Integration Service name>&sqlsds=<runtime SQL data servicename>
User Name	Informatica domain user name.
Password	Informatica domain user password.

8. Click **Test**.  
Squirrel SQL Client displays a message saying that the connection is successful.
9. Click **OK**.

# Tableau Configuration

You can access the virtual data in an SQL data service through Tableau. Tableau uses the ODBC driver to read source data from an SQL data service.

1. Right-click Tableau and choose **Run as Administrator**.
2. Click **Connect to data**.
3. Click **Other Databases (ODBC)**.
4. Select **DSN** to use an existing ODBC connection or select **Driver** to provide the credentials to connect to the SQL data service using the ODBC driver.

If you select **Driver**, provide the connection information to connect to an SQL data service. Tableau saves the credentials and options in the Tableau Workbook (.twb) file when you save the report. By default, the .twb files are located in the following directory `C:\Users\<username>\ Documents\My Tableau Repository\Workbooks`.

5. Click **Connect**.
6. Use Tableau to create a report from an SQL data service.
7. If you need to drag and drop date or numeric fields in Tableau, make the following modifications on the Tableau Workbook file:
  - a. Locate the line `<connection-customization class='genericodbc' enabled='true' version='8.1'>`, and verify that `enabled='true'`.
  - b. Modify the `SQL_NUMERIC_FIELDS` field. Change the value to 6029280: `<customization name='SQL_MAX_IDENTIFIER_LEN' value='0' /> <customization name='SQL_NUMERIC_FUNCTIONS' value='6029280' /> <customization name='SQL_ODBC_INTERFACE_CONFORMANCE' value='1' />`

For more information about Tableau customization, see the Tableau documentation.

## RELATED TOPICS:

- [“Configure ODBC Connections on Windows” on page 55](#)

# Toad for Data Analysts Configuration

Toad for Data Analysts is a database query tool. You can access the virtual data in an SQL data service with Toad for Data Analysts. Use the Navigation Manager in Toad for Data Analysts to maintain and create database connections.

If you connect to one or more databases, use the 'Connections' toolbar to specify the active connection to assign the database connection Toad uses for any new windows or operations.

To configure Toad for Data Analysts to access an SQL data service, complete the following tasks:

1. Select the new connection option in the toolbar.  
The **Create New Connection** dialog box appears.
2. Select **Generic ODBC** from the **Group** list box.
3. Click in the **Data Source Name** field.
4. Click **Add**.

5. Select the ODBC driver from the list, and click **Finish**.
6. Specify the configuration properties required for the database in the windows configuration dialog box.

The following table describes the configuration properties that you can specify:

Option	Description
Data Source Name	Select the ODBC data source name for the SQL data service.
User ID	Informatica domain user name.
Password	Informatica domain user password.

7. Specify the connection properties in the **Create New Connection** dialog.

The following table describes the connection properties that you can specify:

Option	Description
Use data source name	Select to display the Data source names. Deselect to display the driver names.
Data source name	Select the data source name you added in the previous steps.
User	Enter the user name to use when connecting.
Password	Enter the password to use when connecting.
Database	Specify the location of the database schema.
Data Source Name	Displays the selected data source name.
Driver	Displays the ODBC driver associated with the data source.
Category	Select or create a category if you want to color code Editor tabs for a specific connection. This can help differentiate between development and production databases. You can also set an option to color code the <b>Object Explorer</b> pane and object editor windows .

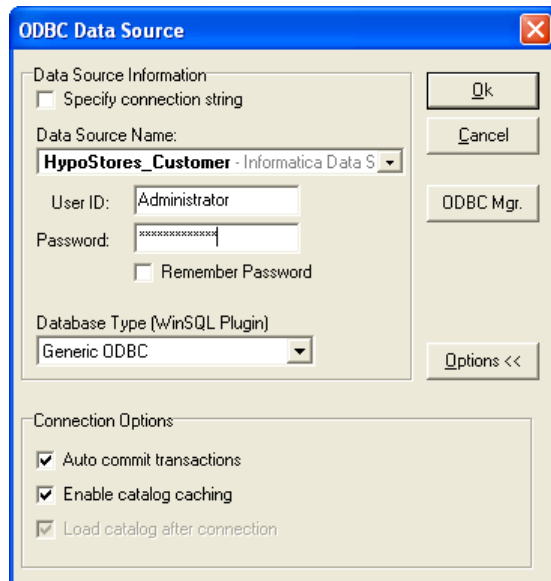
8. Click **Connect** to save the connection and immediately connect to the database, or click **Save** to save the connection without connecting to the database.

## WinSQL Configuration

You can access the virtual data in an SQL data service through WinSQL. To read data from an SQL data service into WinSQL, create a new connection. WinSQL imports data from the SQL data service based on the connection information.

1. Create a query.
2. Select **File > New Connection**.

The **ODBC Data Source** dialog box appears.



3. Enter the ODBC data source properties.

The following table describes the ODBC data source properties that you can enter:

Option	Description
Data Source Name	Select the ODBC data source name for the SQL data service.
User ID	Informatica domain user name.
Password	Informatica domain user password.
Database Type (WinSQL Plugin)	Select <b>Generic ODBC</b> .

4. Click **OK**.

## Troubleshooting Third-Party Client Tools

An error occurs when I query an SQL data service using Oracle Heterogeneous Services (OHS).

If you use Oracle Heterogeneous Services (OHS) to access data services directly from an Oracle query, add quotation marks around the database link when you configure the Oracle 11g environment to connect to the ODBC.

You can use the following syntax to configure the database link in an Oracle 11g environment:

```
CREATE DATABASE LINK "DataService_Link"  
CONNECT TO "userID" IDENTIFIED BY "password"  
USING 'DataService ODBC DSN';
```



For example:

```
CREATE DATABASE LINK "IDS_LINK1"  
CONNECT TO "Admin" IDENTIFIED BY "Admin1234"  
USING 'ids1';
```

An SQL data service is case-sensitive. When you query an SQL data service, the virtual schema and table name must be identical to the names defined in the SQL data service.

#### **An error occurs when I test a new ODBC connection through the Informatica Data Services ODBC Driver:**

```
[SQLCMN_10007] The SQL Service Module could not find an SQL data service on the server  
with the name [<SQL data service name>]. Check the SQL data service name.
```

When you enter the SQLData Service Name, use the correct syntax. This is the correct syntax:

```
<application>.<SQL data service name>
```

#### **A library error occurs when I create, configure, or test an ODBC connection.**

Verify that the PATH environment variable has not exceeded the character limit and that it includes the path to the ODBC driver installation directory.

## CHAPTER 6

# Installing and Configuring Drivers for PowerCenter

This chapter includes the following topics:

- [Installing and Configuring Drivers for PowerCenter Overview, 82](#)
- [Before You Install the Drivers, 82](#)
- [Install the Drivers on Windows, 84](#)
- [Configure ODBC Connections on Windows, 85](#)
- [Install the Drivers on UNIX, 86](#)
- [Configure ODBC Connections on UNIX, 88](#)
- [Troubleshooting ODBC Connections for PowerCenter, 89](#)

## Installing and Configuring Drivers for PowerCenter Overview

You can connect to an SQL data service from PowerCenter®. Before you connect from PowerCenter, you must install and configure the drivers on both the PowerCenter Client and PowerCenter Integration Service machines. After you install and configure the drivers, you can import the SQL data service into the PowerCenter Client and run sessions with the PowerCenter Integration Service.

Before you can connect to an SQL data service from PowerCenter, you must perform the following tasks:

1. Record the Informatica domain information.
2. Install or upgrade the Informatica JDBC/ODBC drivers for PowerCenter.
3. Configure the Informatica JDBC/ODBC drivers for PowerCenter.

## Before You Install the Drivers

Before you install the Informatica JDBC/ODBC drivers for PowerCenter, you must extract the installer files. You must also record information about the domain.

## Extract the Installer Files

The installer files are compressed and distributed as a zip file on Windows and as a tar file on UNIX.

Use a zip utility, a native tar utility, or a GNU tar utility to extract the installer files to a directory on your machine.

You can extract the installer files in the following ways:

- Installation DVD. Download the Informatica zip or tar file from the installation DVD to a directory on your machine and then extract the installer files. Or, extract the installer files directly from the DVD to a directory on your machine.
- FTP download. Download the Informatica installation zip or tar file from the Informatica Electronic Software Download site to a directory on your machine and then extract the installer files.

## Record Informatica Domain Information

When you install and configure the drivers to access an SQL data service, you need to know information about the domain. If you do not have the domain information, contact the Informatica domain administrator.

The following table describes information that you need to know about the domain:

Required Information	Description
Data Integration Service name	Data Integration Service that runs the application that contains the SQL data service.
Domain host name	Machine that hosts the Informatica domain.
Domain HTTP port	Informatica domain HTTP port number.
SQL data service name	Name of the SQL data service that contains the virtual tables you want to query or the virtual stored procedures that you want to run. The run-time SQL data service name includes the application name that contains the SQL data service and uses the following format: <application name>.<SQL data service name>
Security domain name	Informatica security domain name. Required if the Informatica user account is in an LDAP security domain.
User name	Informatica domain user name.
User password	Informatica domain user password.

Required Information	Description
Truststore file	If the Informatica domain has secure communication enabled, you must have the location of the truststore file that contains the SSL certificate for the domain.
Authentication type	<p>The mode of authentication used to connect to the SQL data service. You can select one of the following authentication modes:</p> <p><b>Native or LDAP Authentication</b></p> <p>Uses an Informatica domain user account to connect to the SQL data service in an Informatica domain that uses Native or LDAP authentication. The user account can be in a native or LDAP security domain.</p> <p><b>Kerberos with keytab</b></p> <p>Uses the service principal name (SPN) of an Informatica domain user account to connect to the SQL data service in an Informatica domain that uses Kerberos authentication.</p> <p><b>Kerberos with user name and password</b></p> <p>Uses an Informatica domain user account to connect to the SQL data service in an Informatica domain that uses Kerberos authentication.</p> <p><b>Logged in user</b></p> <p>Uses the user account logged in to the client machine to connect to the SQL data service in an Informatica domain that uses Native, LDAP, or Kerberos authentication.</p>

## Install the Drivers on Windows

Before you can connect to an SQL data service from PowerCenter, you must install or upgrade the Informatica Data Services JDBC/ODBC drivers for PowerCenter. Install the drivers on the machine that you connect to the SQL data service from. You can install the drivers on multiple machines in graphical mode or silent mode.

The Informatica driver version must match the Informatica version.

### Install the Drivers for PowerCenter in Graphical Mode

To install or upgrade the drivers in graphical mode, run the Informatica JDBC/ODBC driver for PowerCenter installer.

1. Close all other applications.
2. Navigate to the root directory of the extracted installer files.
3. Run `install.bat`.
4. Choose to install or upgrade the drivers.
5. Click **Next**.

The **Installation Prerequisites** page displays the system requirements. Verify that all installation requirements are met before you continue the installation.

6. Click **Next**.

7. On the **Installation Directory** page, enter the absolute path for the installation directory.
8. On the **Pre-Installation Summary** page, review the installation information, and click **Install**.  
The installer copies the driver files to the installation directory. The **Post-Installation Summary** page indicates whether the installation completed successfully.

9. Click **Done**.

You can view the installation log files to get more information about the tasks performed by the installer.

## Install the Drivers for PowerCenter in Silent Mode

To install or upgrade the drivers without user interaction, install in silent mode. Use a properties file to specify the installation options. The driver installer reads the file to determine the installation options.

Use the silent mode installation to install the drivers on multiple machines on the network or to standardize the installation across machines.

1. Use a text editor to open and change the values of the properties in the file.

The following table describes the installation properties that you can change:

Property	Description
INSTALL_TYPE	Indicates whether to install or upgrade the drivers. <ul style="list-style-type: none"><li>• Enter 0 to install the drivers.</li><li>• Enter 1 to upgrade the drivers.</li></ul> Default is 0.
USER_INSTALL_DIR	Informatica driver installation directory. For example, C:\\Informatica\\<version>.

2. Save the properties file.
3. Navigate to the root directory of the extracted installer files.
4. To run the silent installation, double-click `silentInstall.bat`.

The silent installer runs in the background. The silent installation is complete when the `Informatica_<Version>_Driver_InstallLog.log` file is created in the root directory.

The silent installation fails if you incorrectly configure the properties file or if the installation directory is not accessible. View the installation log file in `SYSTEM_DRIVE_ROOT`. For example, C:\\`silentErrorLog.log`. Correct the errors, and then run the silent installation again.

## Configure ODBC Connections on Windows

Before you can connect to an SQL data service on Windows, you must configure the Informatica ODBC driver.

1. Open the **Administrative Tools** from the Windows Control Panel.
2. Open the **Data Sources (ODBC)** shortcut.  
The **ODBC Data Source Administrator** appears.
3. Click **Add**.
4. Select the **Informatica Data Services ODBC Driver for PowerCenter**.

5. Click **Finish**.
6. Configure the driver options.

The following table describes the driver options and values that you can configure:

Option	Value
DSN Name	Any valid data source name.
Connect String	<p>Connection string for the JDBC connection.</p> <p>The JDBC connection string uses the following syntax:</p> <pre>jdbc:informatica:sqlds/&lt;optional security domain\&gt;&lt;optional user name&gt;/&lt;optional user password&gt;@&lt;domain host name&gt;:&lt;domain HTTP port&gt;?dis=&lt;Data Integration Service name&gt;&amp;sqlds=&lt;run-time SQL data service name&gt;</pre>
Location for INFADSJDBC.JAR	Path and file name of infadsjdbc.jar. Click Browse to select the jar file for the driver. By default, the jar file is installed in the following directory: <Informatica installation directory>\tools\jdbcdrv
JVM Options	<p>Optional. JVM parameters that you can set to configure the JDBC connection. Use the following arguments to configure the parameters:</p> <ul style="list-style-type: none"> <li>- java -Xms&lt;size&gt;. Sets the initial Java heap size.</li> <li>- java -Xmx&lt;size&gt;. Sets the maximum Java heap size.</li> </ul> <p>For example, java -Xmx2048m -Xms256m starts the JVM with 256 MB of memory, and allows the process to use up to 2048 MB of memory.</p>
Treat Length as Characters (Deferred Parameters)	Disabled.
Multithreaded application	Enabled.

7. Click **Test Connection** to verify that the connection is valid and then click **OK**.

## Install the Drivers on UNIX

Before you can connect to an SQL data service from PowerCenter, you must install or upgrade the Informatica Data Services JDBC/ODBC drivers for PowerCenter. Install the drivers on the machine that you connect to the SQL data service from. You can install the drivers on multiple machines in console or silent mode.

Before you can connect to an SQL data service on UNIX, you must perform the following tasks:

1. Install or upgrade the Informatica JDBC/ODBC drivers for PowerCenter.
2. Configure the shared library environment variable.
3. Configure the `odbc.ini` file.

## Install the Drivers for PowerCenter in Console Mode

To install or upgrade the drivers in console mode, run the Informatica JDBC/ODBC driver for PowerCenter installer.

1. Log in to the machine with a system user account.
2. Close all other applications.
3. Navigate to the root directory of the extracted installer files.
4. At the command prompt, extract the install media.
5. Run `install.sh`.
6. Select one of the following options:

Option	Description
1	Install the drivers.
2	Upgrade the drivers.

7. Press **Enter**.  
The Installation Prerequisites section displays the system requirements.
8. Verify that all installation requirements are met before you continue the installation.
9. Press **Enter**.  
The Installation Directory section appears.
10. Enter the absolute path for the installation directory.
11. Press **Enter**.
12. In the Pre-Installation Summary section, review the installation information, and then press **Enter**.
13. Press **Enter**.

For more information about the install tasks, see the installation debug log.

## Install the Drivers for PowerCenter in Silent Mode

To install or upgrade the drivers without user interaction, install in silent mode. Use a properties file to specify the installation options. The driver installer reads the file to determine the installation options.

Use the silent mode installation to install the drivers on multiple machines on the network or to standardize the installation across machines.

1. Use a text editor to open and change the values of the properties in the file.

The following table describes the installation properties that you can change:

Property	Description
INSTALL_TYPE	Indicates whether to install or upgrade the drivers. <ul style="list-style-type: none"><li>• Enter 0 to install the drivers.</li><li>• Enter 1 to upgrade the drivers.</li></ul> Default is 0.
USER_INSTALL_DIR	Informatica driver installation directory. For example, C:\\Informatica\\<version>.

2. Save the properties file.
3. Navigate to the root directory of the extracted installer files.
4. To run the silent installation, double-click `silentInstall.bat`.

The silent installer runs in the background. The silent installation is complete when the `Informatica_<Version>_Driver_InstallLog.log` file is created in the root directory.

The silent installation fails if you incorrectly configure the properties file or if the installation directory is not accessible. View the installation log file in `SYSTEM_DRIVE_ROOT`. For example, C:\\`silentErrorLog.log`. Correct the errors, and then run the silent installation again.

## Configure ODBC Connections on UNIX

If the PowerCenter Integration Service runs sessions against an SQL data service on UNIX, you must update the shared library environment variable and the `odbc.ini` file.

### Configure the Shared Library Environment Variable

Configure the shared library environment variable to include the directories where the Java Virtual Machine and driver manager library files reside.

Configure the shared library environment variable based on the operating system. The following table lists the shared library environment variable for each operating system:

Operating System	Shared Library Environment Variable
AIX	LIBPATH
HP-UX	SHLIB_PATH
Linux	LD_LIBRARY_PATH
Solaris	LD_LIBRARY_PATH

Configure the shared library environment variable to include the following directories:

- The directory where the platform `libjvm` and `j9vm` libraries reside.
- The directory where the driver manager library files reside. Use the DataDirect driver manager. The DataDirect driver manager is found in `ODBCHOME/lib`.



Use the DataDirect driver manager to create an ODBCINST environment variable to point to the `odbcinst.ini` file.

## Configure the ODBC Files

Before you can connect to an SQL data service on UNIX, you must configure the `odbc.ini` and `odbcinst.ini` files.

Store the `odbcinst.ini` file in any location. Use the `${ODBCINST}` environment variable to point to the `odbcinst.ini` file.

1. Edit `odbc.ini` or copy `odbc.ini` to the home directory and edit it.

This file exists in the `$ODBCHOME` directory.

```
$ cp $ODBCHOME/odbc.ini $HOME/.odbc.ini
```

2. Add an entry for the ODBC user under the section `[<user name>_odbc]`.

For example:

```
[<user name>_odbc]
ConnectionString=jdbc:informatica:sqls/<optional security domain><user name>/<user
password>@<domain host name>:<domain HTTP port>?dis=<Data Integration Service
name>&sqls=<run-time SQL data service name>&authType=<type>
Driver=$ODBC_DRIVER_INSTALL_LOCATION/odbcdrv/libinfadsodbc.so
IsMultiThreaded=true
JDBCDriverLocation=<Informatica installation directory>/tools/jdbcdrv/infadsjdbc.jar
UseDetach=false
```

The Authentication Mode parameter can have one of the following values:

- `native_uid`
- `kerberos_keytab`
- `kerberos_uid`
- `sso`

3. Add the driver and setup entries to the `odbcinst.ini` file.

```
[Informatica Data Services ODBC Driver for PowerCenter <version>]
Driver=<Complete path to driver>
Setup=<Complete path to driver>
```

For example:

```
Driver=$ODBC_DRIVER_INSTALL_LOCATION/odbcdrv/libinfadsodbc.so
Setup=$ODBC_DRIVER_INSTALL_LOCATION/odbcdrv/libinfadsodbc.so
```

## Troubleshooting ODBC Connections for PowerCenter

The ODBC application consumes large amounts of data.

Increase the maximum amount of memory available to the Java Virtual Machine (the `-Xmx` value).

To increase the `-Xmx` value, set the environment variable `INFA_ODBCJVM` to `-Xmx<megabytes>m`. For example, to set the `-Xmx` value to 64 MB, set `INFA_ODBCJVM` to `-Xmx 64m`. If you set the `-Xmx` value to a very large value, for example `>500 MB`, the Memory Manager may not be able to allocate the memory.

### A mapping that contains a Lookup transformation fails.

When a mapping that was created in PowerCenter contains a lookup to an SQL data service through ODBC, and the Lookup transformation performs the lookup before the mapping executes Java transformations, the Lookup will fail. The reason is that when the mapping connects through ODBC to run the Lookup, the connection creates a JVM. This JVM lacks knowledge of the classpath that the later Java transformation needs for its connection.

To work around this issue, enable the PowerCenter ODBC driver.

To do this, you must edit the `infaservice.bat` or `infaservice.sh` file to set the classpath. For example:

```
set CLASSPATH=D:\Informatica\961hf1\1007\server\source\server\bin\javalib
\pmserver sdk.jar;D:\Informatica\961hf1\1007\server\source\server\bin\javalib\pmjtx.jar
call "%CATALINA_HOME%\bin\setclasspath.bat"
set CLASSPATH=%INFA_JAVA_CMD_CLASSPATH%;%CLASSPATH%;%JAVA_HOME%\lib\tools.jar;
%CATALINA_HOME%\bin\bootstrap.jar;%CATALINA_HOME%\bin\tomcat-juli.jar;%INFA_HOME%
\services\shared\jars\platform\infatomcatbootstrap.jar;
```

After you edit the classpath, restart the domain for your changes to take effect.

### The Informatica Data Services ODBC/JDBC driver for PowerCenter does not support single sign-on based authentication when the driver is used with PowerCenter clients to import virtual tables in Informatica domains that use Kerberos authentication.

You must provide the keytab or username and password to import virtual tables and run workflows with the ODBC/JDBC driver and PowerCenter client.

## CHAPTER 7

# SQL Data Service Administration

This chapter includes the following topics:

- [SQL Data Service Administration Overview, 91](#)
- [SQL Data Service Security Management, 92](#)
- [SQL Data Service Properties Configuration , 94](#)
- [SQL Data Service Result Set Caching, 100](#)
- [Data Object Caching, 101](#)
- [SQL Data Service Logs, 102](#)
- [Monitor SQL Data Services, 102](#)

## SQL Data Service Administration Overview

After you deploy an SQL data service to a Data Integration Service, you can configure the service, assign users permissions for the service, view logs, and monitor service requests. You must have the appropriate privileges in order to perform these tasks.

After you deploy an SQL data service to a Data Integration Service, you can complete the following tasks:

- Assign permissions. Enable SQL data service security and assign permissions on SQL data service objects.
- Configure the SQL data service. Configure read-only general properties, Data Integration Service settings, logical data object, and caching properties.
- View the SQL data service logs. View Data Integration Service logs for an SQL data service.
- Monitor the SQL data service. Use the Administrator tool or the Monitoring tool to monitor SQL data service requests.

# SQL Data Service Security Management

You manage user security with privileges and permissions. Permissions define the level of access users have to an SQL data service. You can deny access to columns and rows in a virtual table. You can use credentials to restrict access to data in an SQL data service.

You can assign permissions to users and groups on the following SQL data service objects:

- SQL data service
- Virtual table
- Virtual stored procedure

When you assign permissions on an SQL data service object, the user or group inherits the same permissions on all objects that belong to the SQL data service object. For example, you assign a user select permission on an SQL data service. The user inherits select permission on all virtual tables in the SQL data service.

You can deny permissions to users and groups on some SQL data service objects. When you deny permissions, you configure exceptions to the permissions that users and groups might already have. For example, you cannot assign permissions to a column in a virtual table, but you can deny a user from running an SQL SELECT statement that includes the column.

You can restrict access to specific columns and rows to prevent users from accessing data in an SQL data service when they query a virtual table. Configure column level security to restrict access to specific columns in a virtual table. Configure row level security to restrict access to specific rows of data in a virtual table.

Apply pass-through security to restrict access to data in an SQL data service based on user credentials.

## Types of SQL Data Service Permissions

You can assign the following permissions to users and groups:

- Grant permission. Users can grant and revoke permissions on the SQL data service objects using the Administrator tool or using the *infacmd* command line program.
- Execute permission. Users can run virtual stored procedures in the SQL data service using a JDBC or ODBC client tool.
- Select permission. Users can run SQL SELECT statements on virtual tables in the SQL data service using a JDBC or ODBC client tool.

Some permissions are not applicable for all SQL data service objects.

The following table describes the permissions for each SQL data service object:

Object	Grant Permission	Execute Permission	Select Permission
SQL data service	Grant and revoke permission on the SQL data service and all objects within the SQL data service.	Run all virtual stored procedures in the SQL data service.	Run SQL SELECT statements on all virtual tables in the SQL data service.
Virtual table	Grant and revoke permission on the virtual table.	-	Run SQL SELECT statements on the virtual table.
Virtual stored procedure	Grant and revoke permission on the virtual stored procedure.	Run the virtual stored procedure.	-

## Column Level Security

Column level security is the ability to deny access to individual columns in a virtual table. When end users query columns that they are not authorized to view, the Data Integration Service returns substitute data values, null values, or an error.

An administrator can deny access to columns in a virtual table of an SQL data object. The administrator can configure the Data Integration Service behavior for queries against a restricted column.

The following results might occur when the user queries a column that the user does not have permissions for:

- The query returns a substitute value instead of the data. The query returns a substitute value in each row that it returns. The substitute value replaces the column value through the query. If the query includes filters or joins, the results substitute appears in the results.
- The query fails with an insufficient permission error.

## Row Level Security

Row level security is a level of security that restricts rows of data from users or user groups when they query a virtual table. An administrator can create a security predicate that limits query access to specific rows of data.

A security predicate is an SQL `WHERE` clause that filters data out of the result set when you query a virtual table. The Data Integration Service modifies the query based on security predicates.

For example, a financial services company needs to integrate order data that is stored in disparate data sources and provide a single, integrated orders view. An administrator can create a virtual table that combines the order data and then restrict access to the table with security predicates so that users and groups see a particular data set when they query the table. Employees can access data for orders that they own and for orders in their region within a specified dollar amount.

## Pass-Through Security

Pass-through security is the capability to connect to an SQL data service with the client user credentials instead of the credentials from a connection object.

Users might have access to different sets of data based on the job in the organization. Client systems restrict access to databases by the user name and the password. When you create an SQL data service, you might combine data from different systems to create one view of the data. However, when you define the connection to the SQL data service, the connection has one user name and password.

If you configure pass-through security, you can restrict users from some of the data in an SQL data service based on their user name. When a user connects to the SQL data service, the Data Integration Service ignores the user name and the password in the connection object. The user connects with the client user name or the LDAP user name.

Configure pass-through security for a connection in the connection properties of the Administrator tool or with `infacmd dis UpdateServiceOptions`. You can set pass-through security for connections to deployed applications. You cannot set pass-through security in the Developer tool. Only SQL data services and web services recognize the pass-through security configuration.

For more information about configuring security for SQL data services, see the Informatica How-To Library article "How to Configure Security for SQL Data Services":

[https://kb.informatica.com/h2l/HowTo%20Library/1/0266\\_ConfiguringSecurityForSQLDataServices.pdf](https://kb.informatica.com/h2l/HowTo%20Library/1/0266_ConfiguringSecurityForSQLDataServices.pdf).

## Pass-Through Security with Data Object Caching

To use data object caching with pass-through security, you must enable caching in the pass-through security properties for the Data Integration Service.

When you deploy an SQL data service or a web service, you can choose to cache the logical data objects in a database. You must specify the database in which to store the data object cache. The Data Integration Service validates the user credentials for access to the cache database. If a user can connect to the cache database, the user has access to all tables in the cache. The Data Integration Service does not validate user credentials against the source databases when caching is enabled.

For example, you configure caching for the EmployeeSQLDS SQL data service and enable pass-through security for connections. The Data Integration Service caches tables from the Compensation and the Employee databases. A user might not have access to the Compensation database. However, if the user has access to the cache database, the user can select compensation data in an SQL query.

When you configure pass-through security, the default is to disallow data object caching for data objects that depend on pass-through connections. When you enable data object caching with pass-through security, verify that you do not allow unauthorized users access to some of the data in the cache. When you enable caching for pass-through security connections, you enable data object caching for all pass-through security connections.

## Pass-through Security Properties

The following table describes the pass-through security properties:

Property	Description
Allow Caching	Allows data object caching for all pass-through connections in the Data Integration Service. Populates data object cache using the credentials from the connection object. <b>Note:</b> When you enable data object caching with pass-through security, you might allow users access to data in the cache database that they might not have in an uncached environment.

# SQL Data Service Properties Configuration

Configure the SQL data service properties for each SQL data service that you deploy to a Data Integration Service.

To view or edit the properties of an SQL data service in the Administrator tool, select the Applications view of the Data Integration Service, expand the application name in the top panel, and select the SQL data service. The properties appear in the Properties view.

## SQL Data Service Properties

SQL data service properties include read-only general properties and properties to configure the settings the Data Integration Service uses when it runs the SQL data service.

When you expand an SQL data service in the top panel of the Applications view, you can access the following objects contained in an SQL data service:

- Virtual tables

- Virtual columns
- Virtual stored procedures

The Applications view displays read-only general properties for SQL data services and the objects contained in the SQL data services. Properties that appear in the view depend on the object type.

The following table describes the read-only general properties for SQL data services, virtual tables, virtual columns, and virtual stored procedures:

Property	Description
Name	Name of the selected object. Appears for all object types.
Description	Short description of the selected object. Appears for all object types.
Type	Type of the selected object. Appears for all object types.
Location	The location of the selected object. This includes the domain and Data Integration Service name. Appears for all object types.
JDBC URL	JDBC connection string used to access the SQL data service. The SQL data service contains virtual tables that you can query. It also contains virtual stored procedures that you can run. Appears for SQL data services.
Column Type	Datatype of the virtual column. Appears for virtual columns.

The following table describes the configurable SQL data service properties:

Property	Description
Startup Type	Determines whether the SQL data service is enabled to run when the application starts or when you start the SQL data service. Enter ENABLED to allow the SQL data service to run. Enter DISABLED to prevent the SQL data service from running.
Trace Level	Level of error written to the log files. Choose one of the following message levels: <ul style="list-style-type: none"> <li>- OFF</li> <li>- SEVERE</li> <li>- WARNING</li> <li>- INFO</li> <li>- FINE</li> <li>- FINEST</li> <li>- ALL</li> </ul> Default is INFO.
Connection Timeout	Maximum number of milliseconds to wait for a connection to the SQL data service. Default is 3,600,000.
Request Timeout	Maximum number of milliseconds for an SQL request to wait for an SQL data service response. Default is 3,600,000.
Sort Order	Sort order that the Data Integration Service uses for sorting and comparing data when running in Unicode mode. You can choose the sort order based on your code page. When the Data Integration runs in ASCII mode, it ignores the sort order value and uses a binary sort order. Default is binary.
Maximum Active Connections	Maximum number of active connections to the SQL data service.

Property	Description
Result Set Cache Expiration Period	The number of milliseconds that the result set cache is available for use. If set to -1, the cache never expires. If set to 0, result set caching is disabled. Changes to the expiration period do not apply to existing caches. If you want all caches to use the same expiration period, purge the result set cache after you change the expiration period. Default is 0.
DTM Keep Alive Time	Number of milliseconds that the DTM instance stays open after it completes the last request. Identical SQL queries can reuse the open instance. Use the keep alive time to increase performance when the time required to process the SQL query is small compared to the initialization time for the DTM instance. If the query fails, the DTM instance terminates.  Must be an integer. A negative integer value means that the DTM Keep Alive Time for the Data Integration Service is used. 0 means that the Data Integration Service does not keep the DTM instance in memory. Default is -1.
Optimization Level	The optimizer level that the Data Integration Service applies to the object. Enter the numeric value that is associated with the optimizer level that you want to configure. You can enter one of the following numeric values: <ul style="list-style-type: none"> <li>- 0. The Data Integration Service does not apply optimization.</li> <li>- 1. The Data Integration Service applies the early projection optimization method.</li> <li>- 2. The Data Integration Service applies the early projection, early selection, push-into, and predicate optimization methods.</li> <li>- 3. The Data Integration Service applies the cost-based, early projection, early selection, push-into, predicate, and semi-join optimization methods.</li> </ul>

## SQL Properties

The following table describes the SQL properties for the Data Integration Service:

Property	Description
DTM Keep Alive Time	Number of milliseconds that the DTM instance stays open after it completes the last request. Identical SQL queries can reuse the open instance. Use the keep alive time to increase performance when the time required to process the SQL query is small compared to the initialization time for the DTM instance. If the query fails, the DTM instance terminates.  Must be greater than or equal to 0. 0 means that the Data Integration Service does not keep the DTM instance in memory. Default is 0.  You can also set this property for each SQL data service that is deployed to the Data Integration Service. If you set this property for a deployed SQL data service, the value for the deployed SQL data service overrides the value you set for the Data Integration Service.
Table Storage Connection	Relational database connection that stores temporary tables for SQL data services. By default, no connection is selected.



Property	Description
Maximum Memory Per Request	<p>The behavior of Maximum Memory Per Request depends on the following Data Integration Service configurations:</p> <ul style="list-style-type: none"> <li>- The service runs jobs in separate local or remote processes, or the service property Maximum Memory Size is 0 (default). Maximum Memory Per Request is the maximum amount of memory, in bytes, that the Data Integration Service can allocate to all transformations that use auto cache mode in a single request. The service allocates memory separately to transformations that have a specific cache size. The total memory used by the request can exceed the value of Maximum Memory Per Request.</li> <li>- The service runs jobs in the Data Integration Service process, and the service property Maximum Memory Size is greater than 0. Maximum Memory Per Request is the maximum amount of memory, in bytes, that the Data Integration Service can allocate to a single request. The total memory used by the request cannot exceed the value of Maximum Memory Per Request.</li> </ul> <p>Default is 50,000,000.</p>
Skip Log Files	Prevents the Data Integration Service from generating log files when the SQL data service request completes successfully and the tracing level is set to INFO or higher. Default is false.

The following table describes the SQL properties for the Data Integration Service process:

Property	Description
Maximum # of Concurrent Connections	Limits the number of database connections that the Data Integration Service can make for SQL data services. Default is 100.

## Virtual Table Properties

Configure whether to cache virtual tables for an SQL data service and configure how often to refresh the cache. You must disable the SQL data service before configuring virtual table properties.

The following table describes the configurable virtual table properties:

Property	Description
Enable Caching	Cache the virtual table in the data object cache database.
Cache Refresh Period	Number of minutes between cache refreshes.
Cache Table Name	<p>The name of the user-managed table from which the Data Integration Service accesses the virtual table cache. A user-managed cache table is a table in the data object cache database that you create, populate, and manually refresh when needed.</p> <p>If you specify a cache table name, the Data Object Cache Manager does not manage the cache for the object and ignores the cache refresh period.</p> <p>If you do not specify a cache table name, the Data Object Cache Manager manages the cache for the object.</p>

## Virtual Column Properties

Configure the properties for the virtual columns included in an SQL data service.

The following table describes the configurable virtual column properties:

Property	Description
Create Index	Enables the Data Integration Service to generate indexes for the cache table based on this column. Default is false.
Deny With	When you use column level security, this property determines whether to substitute the restricted column value or to fail the query. If you substitute the column value, you can choose to substitute the value with NULL or with a constant value. Select one of the following options: <ul style="list-style-type: none"><li>- ERROR. Fails the query and returns an error when an SQL query selects a restricted column.</li><li>- NULL. Returns a null value for a restricted column in each row.</li><li>- VALUE. Returns a constant value for a restricted column in each row.</li></ul>
Insufficient Permission Value	The constant that the Data Integration Service returns for a restricted column.

## Virtual Stored Procedure Properties

Configure the property for the virtual stored procedures included in an SQL data service.

The following table describes the configurable virtual stored procedure property:

Property	Description
Result Set Cache Expiration Period	The number of milliseconds that the result set cache is available for use. If set to -1, the cache never expires. If set to 0, result set caching is disabled. Changes to the expiration period do not apply to existing caches. If you want all caches to use the same expiration period, purge the result set cache after you change the expiration period. Default is 0.

## Logical Data Objects

The Applications view displays logical data objects included in applications that have been deployed to the Data Integration Service.

Logical data object properties include read-only general properties and properties to configure caching for logical data objects.

The following table describes the read-only general properties for logical data objects:

Property	Description
Name	Name of the logical data object.
Description	Short description of the logical data object.

Property	Description
Type	Type of the object. Valid value is logical data object.
Location	The location of the logical data object. This includes the domain and Data Integration Service name.

The following table describes the configurable logical data object properties:

Property	Description
Enable Caching	Cache the logical data object in the data object cache database.
Cache Refresh Period	Number of minutes between cache refreshes.
Cache Table Name	<p>The name of the user-managed table from which the Data Integration Service accesses the logical data object cache. A user-managed cache table is a table in the data object cache database that you create, populate, and manually refresh when needed.</p> <p>If you specify a cache table name, the Data Object Cache Manager does not manage the cache for the object and ignores the cache refresh period.</p> <p>If you do not specify a cache table name, the Data Object Cache Manager manages the cache for the object.</p>

The following table describes the configurable logical data object column properties:

Property	Description
Create Index	Enables the Data Integration Service to generate indexes for the cache table based on this column. Default is false.

## Logical Data Object/Virtual Table Cache PropertiesLogical Data Object Cache Properties

The following table describes the data object and virtual table cache properties:

Property	Description
Cache Removal Time	The number of milliseconds that the Data Integration Service waits before cleaning up cache storage after a refresh. Default is 3,600,000.
Cache Connection	The database connection name for the database that stores the data object cache. Select a valid connection object name.

Property	Description
Maximum Concurrent Refresh Requests	Maximum number of cache refreshes that can occur at the same time. Limit the concurrent cache refreshes to maintain system resources.
Enable Nested LDO Cache	<p>Indicates that the Data Integration Service can use cache data for a logical data object used as a source or a lookup in another logical data object during a cache refresh. If false, the Data Integration Service accesses the source resources even if you enabled caching for the logical data object used as a source or a lookup.</p> <p>For example, logical data object LD03 joins data from logical data objects LD01 and LD02. A developer creates a mapping that uses LD03 as the input and includes the mapping in an application. You enable caching for LD01, LD02, and LD03. If you enable nested logical data object caching, the Data Integration Service uses cache data for LD01 and LD02 when it refreshes the cache table for LD03. If you do not enable nested logical data object caching, the Data Integration Service accesses the source resources for LD01 and LD02 when it refreshes the cache table for LD03.</p> <p>Default is False.</p>

The following table describes the data object cache properties:

Property	Description
Cache Removal Time	The number of milliseconds that the Data Integration Service waits before cleaning up cache storage after a refresh. Default is 3,600,000.
Cache Connection	The database connection name for the database that stores the data object cache. Select a valid connection object name.

## SQL Data Service Result Set Caching

Result set caching enables the Data Integration Service to use cached results for SQL data service queries. Users that run identical queries in a short period of time may want to use result set caching to decrease the runtime of identical queries.

When you configure result set caching, the Data Integration Service caches the results of the DTM process associated with each SQL data service query. The Data Integration Service caches the results for the expiration period that you configure. When an external client makes the same query or request before the cache expires, the Data Integration Service returns the cached results.

The Result Set Cache Manager creates in-memory caches to temporarily store the results of the DTM process. If the Result Set Cache Manager requires more space than allocated, it stores the data in cache files. The Result Set Cache Manager identifies the cache files by file name and location. Do not rename or move the cache files.

Complete the following steps to configure result set caching for SQL data services in the Administrator tool:

1. Configure the result set cache properties in the Data Integration Service process properties.
2. Configure the cache expiration period in the SQL data service properties.

The Data Integration Service purges result set caches in the following situations:

- When the result set cache expires, the Data Integration Service purges the cache.

- When you restart an application or run the *infacmd dis purgeResultSetCache* command, the Data Integration Service purges the result set cache for objects in the application.
- When you restart a Data Integration Service, the Data Integration Service purges the result set cache for objects in applications that run on the Data Integration Service.
- When you change the permissions for a user, the Data Integration Service purges the result set cache associated with that user.

## Data Object Caching

The Data Integration Service uses data object caching to access pre-built logical data objects. Enable data object caching to increase performance for mappings that include logical data objects. The Data Integration Service uses data object caching to access pre-built logical data objects and virtual tables. Enable data object caching to increase performance for mappings, SQL data service queries, and web service requests that include logical data objects and virtual tables.

By default, the Data Integration Service extracts source data and builds required data objects when it runs a mapping, SQL data service query, or a web service request. When you enable data object caching, the Data Integration Service can use cached logical data objects and virtual tables.

By default, the Data Integration Service extracts source data and builds required data objects when it runs a mapping. When you enable data object caching, the Data Integration Service can use cached logical data objects.

Perform the following steps to configure data object caching for logical data objects and virtual tables in an application:

1. Configure the data object cache database connection in the cache properties for the Data Integration Service.
2. Enable caching in the properties of logical data objects or virtual tables in an application.

Perform the following steps to configure data object caching for logical data objects in an application:

1. Configure the data object cache database connection in the cache properties for the Data Integration Service.
2. Enable caching in the properties of logical data objects in an application.

By default, the Data Object Cache Manager component of the Data Integration Service manages the cache tables for logical data objects and virtual tables in the data object cache database. When the Data Object Cache Manager manages the cache, it inserts all data into the cache tables with each refresh. If you want to incrementally update the cache tables, you can choose to manage the cache tables yourself using a database client or other external tool. After enabling data object caching, you can configure a logical data object or virtual table to use a user-managed cache table.

By default, the Data Object Cache Manager component of the Data Integration Service manages the cache for logical data objects in the data object cache database. When the Data Object Cache Manager manages the cache, it inserts all data into the cache tables with each refresh.

To use the Timestamp with Time Zone data type and to enable data object caching for IBM DB2 or for Microsoft SQL Server, set the date time format of the deployed mapping to the "YYYY-MM-DD HH24:MI:SS" format. The Data Integration Service writes the data up to seconds.

# SQL Data Service Logs

You can view SQL data service logs in the Data Integration Service logs. View the Data Integration Service logs on the **Logs** tab in the Administrator tool.

Data Integration Service logs contain logs about the following events:

- Configuration. Log events about system or service configuration changes and application deployment or removal.
- Data Integration Service processes. Log events about application deployment, data object cache refresh, and user requests to run mappings.
- System failures. Log events about failures that cause the Data Integration service to be unavailable, such as Model Repository connection failures or the service failure to start.

## Monitor SQL Data Services

Monitor an SQL data service to view its properties, connections, requests, virtual tables and reports. You can also view graphical information about the distribution and state of SQL data services across the Data Integration Services.

You can monitor an SQL data service in the following locations:

- Monitoring tool. In the Developer tool **Progress** view, click **Menu > Monitor Jobs**. Select the Data Integration Service that runs the SQL data service and click **OK**. The Monitoring tool opens.
- Administrator tool. To monitor web services in the Administrator tool, click the **Monitor** tab.

When you monitor an SQL data service, you can view summary statistics or execution statistics for the service. The **Summary Statistics** view displays graphical information about SQL data service distribution and state. The **Execution Statistics** view displays information about SQL data services that are deployed in an application.

To monitor an SQL data service, expand an application in the Navigator and select the **SQL Data Services** folder. A list of SQL data services appears in the contents panel. The contents panel shows properties about each SQL data service, such as the name, description, and state.

When you select an SQL data service in the contents panel, the contents panel shows the following views:

- **Properties** view
- **Connections** view
- **Requests** view
- **Virtual Tables** view
- **Reports** view

## Properties View for an SQL Data Service

The **Properties** view shows general properties and run-time statistics for an SQL data service.

When you select an SQL data service in the contents panel of the **Properties** view, you can view the general properties and run-time statistics.

### General Properties for an SQL Data Service

You can view general properties, such as the SQL data service name and the description.

### Statistics for an SQL Data Service

You can view run-time statistics about connections and requests for the SQL data service. Sample statistics include the number of connections to the SQL data service, the number of requests, and the number of aborted connections.

## Connections View for an SQL Data Service

The **Connections** view displays properties about connections from third-party clients. The view shows properties such as the connection ID, state of the connection, connect time, elapsed time, and disconnect time.

When you select a connection in the contents panel, you can abort the connection or access the **Properties** view and **Requests** view in the details panel.

### Properties View

The **Properties** view in the details panel shows the user who is using the connection, the state of the connection, and the connect time.

### Requests View

The **Requests** view in the details panel shows information about the requests for the SQL connection. Each connection can have more than one request. The view shows request properties such as request ID, user name, state of the request, start time, elapsed time, and end time.

## Aborting a Connection

You can abort a connection to prevent it from sending more requests to the SQL data service.

1. Click the **Execution Statistics** view.
2. In the Domain Navigator, expand a Data Integration Service.
3. Expand an application and select **SQL Data Services**.  
The contents panel lists the SQL data services in the application.
4. Select an SQL data service.  
The contents panel displays multiple views for the SQL data service.
5. Click the **Connections** view.  
The contents panel lists connections to the SQL data service.
6. Select a connection.
7. Click **Actions > Abort Selected Connection**.

## Requests View for an SQL Data Service

The **Requests** view displays properties for requests for each SQL connection.

The **Requests** view shows properties about the requests for the SQL connection. Each connection can have more than one request. The view shows request properties such as request ID, connection ID, user name, state of the request, start time, elapsed time, and end time.

Select a request in the contents panel to view additional information about the request in the details panel.

## Aborting an SQL Data Service Connection Request

You can abort an SQL Data Service connection request. You might want to abort a connection request that hangs or that is taking an excessive amount of time to complete.

1. Click the **Execution Statistics** view.
2. In the Domain Navigator, expand a Data Integration Service.
3. Expand an application and select **SQL Data Services**.  
The contents panel displays a list of SQL data services.
4. Select an SQL data service.
5. Click the **Requests** view.  
A list of connection requests for the SQL data service appears.
6. Select a request row.
7. Click **Actions > Abort Selected Request**.

## Viewing Logs for an SQL Data Service Request

You can download the logs for an SQL data service request to view the request details.

1. Click the **Execution Statistics** view.
2. In the Domain Navigator, expand a Data Integration Service.
3. Expand an application and select **SQL Data Services**.  
The contents panel displays a list of SQL data services.
4. Select an SQL data service.
5. Click the **Requests** view.  
A list of requests for the SQL data service appears.
6. Select a request row.
7. Click **Actions > View Logs for Selected Object**.

## Virtual Tables View for an SQL Data Service

The **Virtual Tables** view displays properties about the virtual tables in the SQL data service.

The view shows properties about the virtual tables, such as the name and description. When you select a virtual table in the contents panel, you can view the **Properties** view and **Cache Refresh Runs** view in the details panel.

### Properties View

The **Properties** view displays general information and run-time statistics about the selected virtual table. General properties include the virtual table name and the schema name. Monitoring statistics include the number of request, the number of rows cached, and the last cache refresh time.

### Cache Refresh Runs View

The **Cache Refresh Runs** view displays cache information for the selected virtual table. The view includes the cache run ID, the request count, row count, and the cache hit rate. The cache hit rate is the total number of requests on the cache divided by the total number of requests for the data object.



## Viewing Logs for an SQL Data Service Table Cache

You can download the logs for an SQL data service table cache to view the table cache details.

1. Click the **Execution Statistics** view.
2. In the Domain Navigator, expand a Data Integration Service.
3. Expand an application and select **SQL Data Services**.  
The contents panel displays a list of SQL data services.
4. Select an SQL data service.
5. Click the **Virtual Tables** view.  
A list of virtual tables for the SQL data service appears.
6. Select a table row.  
Details about the selected table appear in the details panel.
7. Select the **Cache Refresh Runs** view.
8. Click **View Logs for Selected Object**.

## Reports View for an SQL Data Service

The **Reports** view shows monitoring reports about the selected SQL data service.

When you monitor an SQL data service, the **Reports** view shows reports about the SQL data service. For example, you can view the Most Active SQL Connections report to determine the SQL connections that received the most connection requests during a specific time period.

# INDEX

## B

BusinessObjects  
  BusinessObjects Configuration [59](#)  
  creating the ODBC connection [59](#)

## C

Cache Connection  
  property [99](#)  
Cache Removal Time  
  property [99](#)  
CLASSPATH  
  updating [52](#)  
client configuration  
  secure domain [46](#)  
configuration  
  client tool configuration [58](#)  
  IBM Cognos [60](#)  
  MicroStrategy [65](#)  
  Oracle Business Intelligence Enterprise Edition 11g [71](#)  
  Oracle Database Gateway [72](#)  
  QlikView [72](#)  
  SQL Server Business Intelligence Development Studio [74](#)  
  Squirrel SQL Client [75](#)  
  third-party client driver configuration [44](#)  
  Toad for Data Analysts [78](#)  
  WinSQL [79](#)  
connecting  
  SQL data service [93](#)  
connections  
  overview [10](#)  
  pass-through security [93](#)  
  protocol [10](#)  
correlated subqueries  
  rules and guidelines for [38](#)  
  SQL data service queries [38](#)

## D

data object cache  
  configuring [101](#)  
  description [101](#)  
  index cache [101](#)  
  properties [99](#)  
  user-managed tables [101](#)  
data object caching  
  with pass-through security [94](#)  
datatypes  
  SQL data service queries [28](#)

## E

Enable Nested LDO Cache  
  property [99](#)  
environment variables  
  INFA\_TRUSTSTORE\_PASSWORD [46](#)  
  INFA\_TRUSTSTORE [46](#)  
escape syntax  
  SQL data service queries [39](#)

## F

functions  
  escape syntax [40](#)  
  SQL data service queries [28](#)

## I

IBM Cognos  
  configuration [60](#)  
  creating the ODBC connection [61](#)  
  updating the configuration file [61](#)  
infadsjdbc.jar  
  configuring [52](#)  
  troubleshooting [52](#)  
infadsjdbcflight.jar  
  configuring [52](#)  
Informatica JDBC driver  
  infadsjdbc.jar [52](#)  
  infadsjdbcflight.jar [52](#)  
Informatica ODBC driver  
  configuring [85](#)

## J

JDBC client connections  
  connecting to an SQL data service [51](#)  
  overview [45](#), [47](#), [82](#), [84](#)  
  troubleshooting [52](#)  
  updating the CLASSPATH [52](#)

## L

LD\_LIBRARY\_PATH  
  updating [88](#)  
LIBPATH  
  updating [88](#)  
logical data objects  
  caching in database [101](#)

## M

- Maximum Active Connections
  - SQL data service property [94](#)
- Maximum Concurrent Connections
  - configuring [96](#)
- Maximum Concurrent Refresh Requests
  - property [99](#)
- MicroStrategy
  - configuration [65](#)
  - configuring SQL generation options [70](#)
  - creating the database instance [65](#)
- monitoring
  - SQL data services [102](#)

## N

- non-correlated subqueries
  - SQL data service queries [37](#)

## O

- ODBC client connections on UNIX
  - overview [48, 86](#)
  - troubleshooting [89](#)
  - updating odbcc.ini [89](#)
  - updating the shared library [88](#)
- ODBC client connections on Windows
  - configuring the driver [85](#)
  - troubleshooting [89](#)
- odbc.ini
  - updating [89](#)
- operators
  - SQL data service queries [28](#)
- Oracle Business Intelligence Enterprise Edition 11g
  - configuration [71](#)
- Oracle Database Gateway
  - configuration [72](#)

## P

- parameterized queries
  - SQL data service queries [39](#)
  - troubleshooting [52](#)
- pass-through security
  - connecting to SQL data service [93](#)
  - enabling caching [94](#)
  - properties [94](#)
  - web service operation mappings [93](#)

## Q

- QlikView
  - configuration [72](#)
- queries
  - SQL data service queries [37](#)

## R

- request timeout
  - SQL data services requests [94](#)
- reserved words
  - SQL data service queries [39](#)

- result set cache
  - configuring [100](#)
  - purging [100](#)
  - SQL data service properties [94](#)
- result set caching
  - virtual stored procedure properties [98](#)
- row level security
  - description [93](#)

## S

- secure domain
  - client configuration [46](#)
- shared library environment variable
  - updating [88](#)
- sort order
  - SQL data services [94](#)
- special characters
  - SQL data service queries [42](#)
- SQL data service
  - permission types [92](#)
  - properties [94](#)
- SQL data service connections
  - overview [10](#)
- SQL data service queries
  - correlated subqueries [38](#)
  - datatypes [28](#)
  - escape syntax [39](#)
  - function escape sequences [40](#)
  - non-correlated subqueries [37](#)
  - operators [28](#)
  - overview [27](#)
  - parameterized queries [39](#)
  - queries [37](#)
  - reserved words [39](#)
  - special characters [42](#)
  - SQL functions [28](#)
  - SQL statements and keywords [36](#)
- SQL data services
  - connecting from JDBC [51](#)
  - creating [13](#)
  - defining [13](#)
  - example [11](#)
    - Informatica ODBC driver
      - installing on Windows [45, 47, 82, 84](#)
  - JDBC client connections [45, 47, 82, 84](#)
  - monitoring [102](#)
  - ODBC client connections on UNIX [48, 86](#)
    - ODBC client connections on Windows
      - installing the driver [45, 47, 82, 84](#)
      - overview [45, 47, 82, 84](#)
  - overview [10, 13](#)
  - previewing data [17](#)
    - SQL data services
      - ODBC client connections on Windows [45, 47, 82, 84](#)
- SQL keywords
  - SQL data service queries [36](#)
- SQL queries
  - previewing data [17](#)
- SQL query plans
  - example [25](#)
  - overview [24](#)
  - viewing [26](#)
- SQL Server Business Intelligence Development Studio
  - configuration [74](#)
- SQL statements
  - SQL data service queries [36](#)

- Squirrel SQL Client
  - configuration [75](#)
  - copying the driver file [75](#)
  - creating the driver and alias [76](#)
- startup type
  - configuring SQL data services [94](#)

## T

- Tableau
  - configuration [78](#)
- temporary tables
  - description [17](#)
  - operations [18](#)
  - rules and guidelines [20](#)
- third-party client tools
  - supported third-party tools [58](#)
- Third-party Client Tools
  - troubleshooting [80](#)
- timeout
  - SQL data service connections [94](#)
- Toad for Data Analysts
  - configuration [78](#)
- troubleshooting
  - converting large numbers to binary [42](#)
  - converting negative values [42](#)
  - Third-party Client Tools [80](#)

## V

- virtual column properties
  - configuring [98](#)
- virtual data
  - overview [12](#)

- virtual stored procedure properties
  - configuring [98](#)
- virtual stored procedures
  - creating [23](#)
  - defining [23](#)
  - overview [22](#)
  - previewing output [24](#)
  - running [36](#)
  - validating [24](#)
- virtual table mappings
  - creating [21](#)
  - defining [21](#)
  - description [20](#)
  - previewing output [22](#)
  - validating [21](#)
- virtual table properties
  - configuring [97](#)
- virtual tables
  - caching in database [101](#)
  - creating from a data object [15](#)
  - creating manually [16](#)
  - data access methods [15](#)
  - defining relationships [17](#)
  - description [14](#)
  - example [11](#)
  - previewing data [17](#)

## W

- web service
  - logs [102](#)
  - property configuration [94](#)
- WinSQL
  - configuration [79](#)