



Informatica® PowerExchange for Siebel
10.2

User Guide for PowerCenter

© Copyright Informatica LLC 2010, 2018

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, PowerCenter, and PowerExchange 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.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

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>. 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>.

The product includes software copyright © 2004-2007, The Dojo Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://dojotoolkit.org/license>.

This product includes ICU software which is copyright International Business Machines Corporation and others. All rights reserved. Permissions and limitations regarding this software are subject to terms available at <http://source.icu-project.org/repos/icu/icu/trunk/license.html>.

This product includes software copyright © 1996-2006 Per Bothner. All rights reserved. Your right to use such materials is set forth in the license which may be found at <http://www.gnu.org/software/kawa/Software-License.html>.

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.

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.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/licence.html>, <http://www.jgraph.com/jgraphdownload.html>, <http://www.jcraft.com/jsch/LICENSE.txt>, http://jotm.objectweb.org/bsd_license.html, <http://www.w3.org/>

Consortium/Legal/2002/copyright-software-20021231; <http://www.slf4j.org/license.html>; <http://nanoxml.sourceforge.net/orig/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://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/tinkerpop/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 products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA 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: 2018-09-28

Table of Contents

Preface	8
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
 Chapter 1: Understanding PowerExchange for Siebel.....	 10
Understanding PowerExchange for Siebel Overview.	10
Understanding Siebel.	10
Siebel Object Model.	11
Siebel Business Components.	13
Siebel Business Service Transformations.	13
Siebel EIM Transformations.	13
Handling Bulk Data Movement in Siebel.	13
PowerCenter Integration and Siebel.	14
PowerCenter Designer and Siebel.	14
PowerCenter Integration Service and Siebel.	15
 Chapter 2: PowerExchange for Siebel Configuration.....	 17
PowerExchange for Siebel Configuration Overview.	17
Configuring PowerExchange for Siebel.	17
Registering the Plug-in.	18
Registering the Plug-in from the Administrator Tool.	18
Registering the Plug-in from the Command Line Program.	18
Configuring Java Properties.	19
Copying Siebel API Libraries.	19
Adding a Filter to Specify the Siebel Repository.	19
Troubleshooting Configuration.	20
 Chapter 3: Siebel Sources.....	 21
Siebel Sources Overview.	21
Working with Siebel Sources.	21
Generating Key Values.	23
Filtering Siebel Sources.	25
Siebel Source Definitions.	26
Importing a Siebel Business Component.	26

Updating Siebel Source Definitions.	28
Reimporting a Siebel Source Definition.	29
Editing a Siebel Source Definition.	29
Chapter 4: Application Source Qualifier for Siebel Sources.	30
Application Source Qualifier for Siebel Sources Overview.	30
Source Filter.	30
Sort Filter.	31
View Mode.	32
Chapter 5: Siebel Targets.	34
Siebel Targets Overview.	34
Working with Siebel Targets.	34
Generating Key Values.	36
Configuring the Update Strategy.	36
Siebel Target Definitions.	38
Importing a Siebel Target Definition.	38
Updating Siebel Target Definitions.	39
Reimporting a Siebel Target Definition.	40
Editing a Siebel Target Definition.	40
Troubleshooting Siebel Targets.	40
Chapter 6: Siebel Business Service Transformation.	41
Siebel Business Service Transformation Overview.	41
Siebel Business Service Transformation Components.	41
Metadata Extensions.	42
Siebel Business Service Transformation Groups and Ports.	42
Using a Siebel Business Service Transformation in a Mapping.	42
Working with Integration Objects.	43
Working with Hierarchy Objects.	44
Working with Strings, Numbers, and Dates.	45
Creating a Siebel Business Service Transformation.	45
Chapter 7: Siebel EIM Read Transformation.	47
Siebel EIM Read Transformation Overview.	47
Siebel EIM Read Transformation Components.	47
Configuring Siebel EIM Read Transformation Properties.	48
Extracting Rows from Siebel EIM Tables.	48
Siebel EIM Read Transformation Groups and Ports.	49
INPUT_STATUS Input Group.	49
CONTROL_FIELDS Input Group.	49
<Siebel EIM Table Name> Output Group.	50
OUTPUT_STATUS Output Group.	51

Using a Siebel EIM Read Transformation in a Mapping.	51
Write Siebel Base Table Data to a Corporate Data Source.	51
Determining the Row-Level Status of Siebel EIM Functions.	52
Creating a Siebel EIM Read Transformation.	53
Understanding the Read Summary for the Siebel EIM Read Transformation.	55
Chapter 8: Siebel EIM Load Transformation.	56
Siebel EIM Load Transformation Overview.	56
Siebel EIM Load Transformation Components.	56
Configuring Siebel EIM Load Transformation Properties.	56
Siebel EIM Load Transformation Groups and Ports.	57
STATUS Output Group.	57
CONTROL_FIELDS Output Group.	57
ERROR Output Group.	57
Using a Siebel EIM Load Transformation in a Mapping.	58
Importing Data from Siebel EIM Tables into Siebel Base Tables.	58
Merging Data from Siebel EIM Tables into Siebel Base Tables.	59
Creating a Siebel EIM Load Transformation.	61
Rules and Guidelines for the Siebel EIM Load Transformation.	62
Understanding the Load Summary for the Siebel EIM Load Transformation.	62
Chapter 9: Siebel EIM Invoker Transformation.	63
Siebel EIM Invoker Transformation Overview.	63
Siebel EIM Invoker Transformation Components.	63
Configuring an IFB File.	64
Siebel EIM Invoker Transformation Groups and Ports.	64
INPUT_STATUS Input Group.	64
OUTPUT_STATUS Output Group.	65
Using a Siebel EIM Invoker Transformation in a Mapping.	65
Rules and Guidelines for the Siebel EIM Invoker Transformation.	66
Creating a Siebel EIM Invoker Transformation.	66
Chapter 10: Siebel Sessions.	67
Configuring Application Connection Objects for Siebel.	67
Configuring Application Connections for Siebel Sources, Targets, and EIM Invoker Transformations.	67
Configuring Application Connection for Siebel EIM Read and Load Transformations.	68
Configuring Sessions for a Siebel Target.	69
Configuring Sessions for a Siebel EIM Read Transformation.	70
Configuring Sessions for a Siebel EIM Load Transformation.	71
Configuring the Load Type for the Siebel EIM Load Transformation.	72
Validating Data against Siebel List of Values.	73
Truncating EIM Tables.	73

Configuring Sessions for a Siebel EIM Invoker Transformation.	74
EIM Tasks and Partitions.	75
EIM Task and Business Service Modes.	75
Creating an EIM Job Parameters File.	76
Error Handling.	76
Error Handling for Fatal Errors.	76
Error Handling for Non-Fatal Errors.	77
Troubleshooting Siebel Sessions.	77
Appendix A: Datatype Reference.	79
Datatypes Overview.	79
Siebel Business Component and Transformation Datatypes.	80
Siebel Business Service Method and Transformation Datatypes.	81
Siebel EIM and Transformation Datatypes.	81
Appendix B: Error Messages.	83
Designer Messages.	83
PowerCenter Integration Service Messages.	83
Appendix C: Glossary.	88
Index.	90

Preface

The *Informatica PowerExchange® for Siebel User Guide for PowerCenter®* provides information about extracting data from a Siebel source and loading data into a Siebel target. The *User Guide* is written for database administrators and developers who are responsible for extracting data from Siebel and loading data to Siebel. This book assumes you have knowledge of Siebel and PowerCenter.

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.

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.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at 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.

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

Understanding PowerExchange for Siebel

This chapter includes the following topics:

- [Understanding PowerExchange for Siebel Overview, 10](#)
- [Understanding Siebel, 10](#)
- [Handling Bulk Data Movement in Siebel, 13](#)
- [PowerCenter Integration and Siebel, 14](#)

Understanding PowerExchange for Siebel Overview

PowerExchange for Siebel integrates with Siebel business components and business services. You can import Siebel business components as sources and targets in mappings, and then use a session to extract data from or write data to a business component. You can also use PowerExchange for Siebel to invoke Siebel business services.

The PowerCenter Integration Service uses Java Bean API to access Siebel business components and to invoke Siebel business services. The PowerCenter Integration Service maintains data integrity while reading data from and writing data to Siebel by maintaining relationships among Siebel business components.

Understanding Siebel

Siebel Enterprise Applications is an integrated suite of enterprise marketing, sales, customer service, and call center management applications. Siebel Enterprise Applications is based on client-server architecture. It can use any of the major relational databases for data storage on the server.

Siebel Enterprise Applications is based on the Siebel Object Model. The Siebel Object Model contains multiple layers that PowerExchange for Siebel interfaces with to retrieve metadata and read and write data.

A Siebel business component is a logical entity that combines columns from base tables, extension tables, and joined tables into a single structure. Siebel tables have columns and indexes that directly correspond to the columns and indexes of physical database tables in the underlying database.

You can import Siebel business components as source and target definitions. You can import a Siebel business service method as a transformation.

The Siebel EIM data integration approach uses Siebel EIM tables, which correspond to Siebel base tables. You can create Siebel EIM transformations to extract bulk data from and load bulk data into Siebel EIM tables and invoke EIM tasks to extract data from or load data into Siebel base tables.

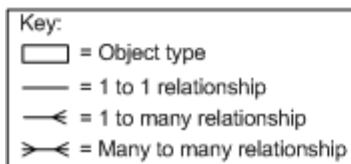
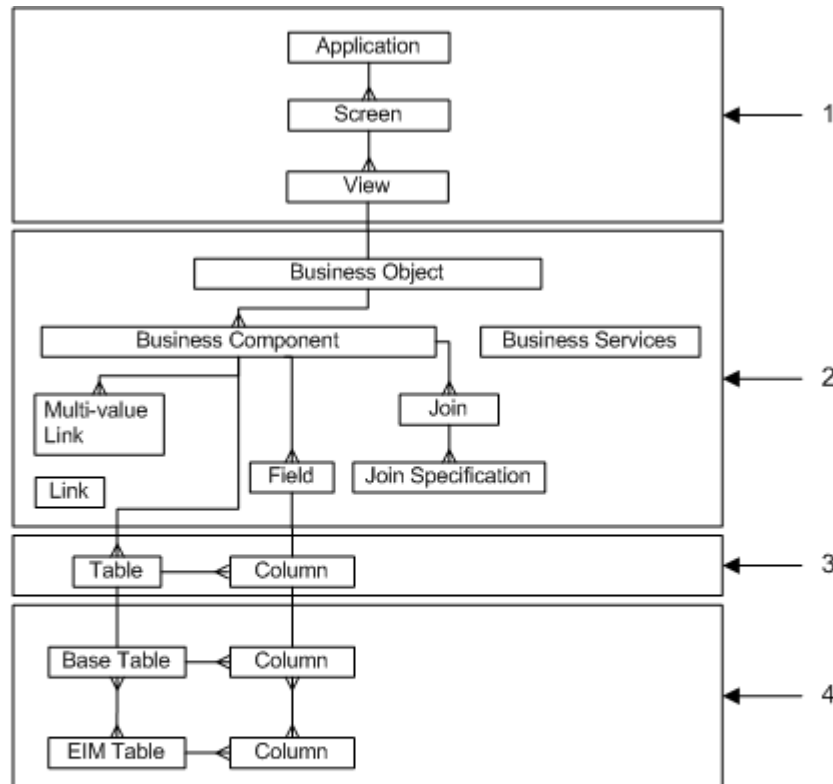
Siebel Object Model

The Siebel Object Model consists of multiple objects. The object definitions in Siebel Enterprise Applications exist in the following architectural layers:

- User Interface Objects Layer. Consists of user interface object definitions. The object definitions describe the interface with which the user interacts in the Siebel Enterprise Applications.
- Business Objects Layer. Consists of business object definitions that are built on data object definitions. Examples of business objects include business components, business services, links, multi-value links, joins, and fields. When the PowerCenter Integration Service writes data to Siebel, it extracts business components and their properties from the Business Objects Layer.
- Data Objects Layer. Consists of data object definitions that correspond to the data structures in the underlying database. Examples of data objects include Siebel tables and columns. This layer is independent of the underlying database.
- Relational Database Management System (RDBMS). Consists of the underlying database tables.

For more information about the Siebel Object Model, see the Siebel documentation.

The following figure shows objects in the Siebel Object Model:



1. User Interface Objects Layer
2. Business Objects Layer
3. Data Objects Layer
4. RDBMS

PowerExchange for Siebel extracts the hierarchy for user interface objects and business objects from Siebel Enterprise Applications. PowerExchange for Siebel extracts definitions for multi-value links, links, fields, joins, and join specifications when it extracts business component metadata. In addition, PowerExchange for Siebel extracts business service metadata.

The following PowerCenter objects connect to Siebel to perform the specified functions:

- Source. Connects to the Siebel repository to extract metadata of Siebel business components and retrieve data from Siebel business components.
- Target. Connects to the Siebel repository to extract metadata of Siebel business components and load data into Siebel business components.
- Siebel Business Service transformation. Connects to the Siebel repository to extract metadata of business service methods and invoke business service methods.

- Siebel EIM Invoker transformation. Connects to the Siebel server to invoke the ServerRequest business service method.
- Siebel EIM Read transformation. Connects to the Siebel repository to extract metadata for EIM tables and connect to Siebel database to extract bulk data from the underlying database tables.

Siebel EIM Load transformation. Connects to the Siebel repository to extract metadata for EIM tables and connect to the Siebel database to load bulk data into the underlying database tables.

Siebel Business Components

A Siebel business component is a logical entity that combines columns from Siebel tables into a single structure. Business components in the Business Objects Layer correspond to the tables on the Data Objects Layer. When you import Siebel objects as PowerCenter objects, PowerCenter retrieves information from multiple layers and objects.

For example, when you import a Siebel business component as a source definition, the Designer imports the metadata from the business components in the Business Objects Layer and the associated Siebel tables in the Data Objects Layer. When you run a session, the PowerCenter Integration Service extracts the application data from the physical database tables in the underlying database.

Siebel Business Service Transformations

You can import a Siebel business service method as a transformation. Siebel business services exist in the Business Objects Layer. You can use a Siebel Business Service transformation to invoke a Siebel business service method that implements business logic.

Siebel EIM Transformations

PowerExchange for Siebel contains multi-group transformations that can extract and load bulk data.

The following table describes the transformations you can use to extract bulk data or load bulk data into Siebel EIM tables:

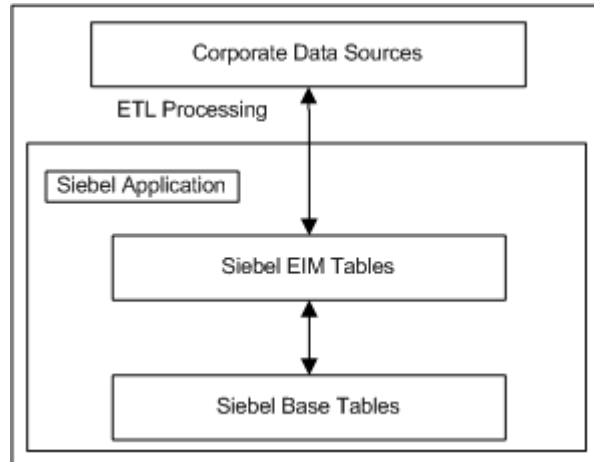
Siebel EIM Transformation	Description
Siebel EIM Read transformation	Extracts bulk data from Siebel EIM tables.
Siebel EIM Load transformation	Loads bulk data into Siebel EIM tables.
Siebel EIM Invoker transformation	Invokes a Siebel EIM task that performs Siebel EIM function on the Siebel server.

Handling Bulk Data Movement in Siebel

Siebel Enterprise Integration Manager (EIM) is a Siebel server component that manages bulk data movement between the Siebel database and the other corporate data sources. Siebel EIM is primarily used to extract bulk data from or load bulk data into the Siebel database. You can use Siebel EIM to perform insert, update, merge, or delete operations for bulk data.

Based on the Siebel architecture, you cannot directly perform operations on the Siebel base tables. Instead, you have to use staging tables, called Siebel EIM tables, to transfer bulk data to and from Siebel tables, known as the Siebel base tables.

The following figure shows the Siebel EIM architecture:



If you configure an EIM task to export bulk data from base tables into EIM tables, Siebel EIM runs the EIM task on the Siebel server. You can use the Siebel EIM Read transformation to extract bulk data from Siebel EIM tables, and then write the data to corporate data sources.

You can use the Siebel EIM Load transformation to load bulk data from corporate data sources into the Siebel EIM tables. Next, you configure an EIM task to import bulk data from EIM tables into base tables. Siebel EIM runs the EIM task on the Siebel server.

To configure an EIM task to import, export, merge, or delete rows from the Siebel base table, configure the EIM configuration (.ifb) file.

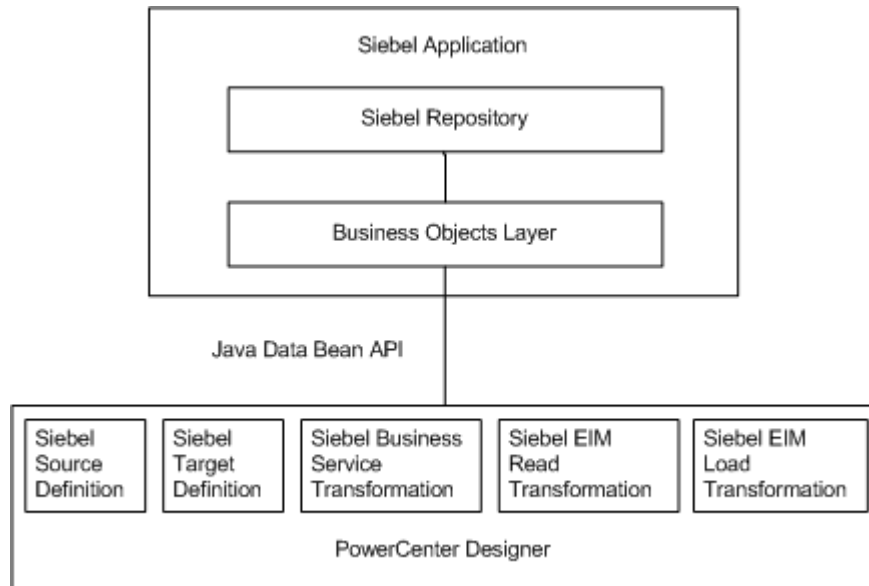
PowerCenter Integration and Siebel

You can create sessions that extract data from Siebel tables, write data to Siebel tables, and invoke Siebel business services and EIM tasks.

PowerCenter Designer and Siebel

When you import a source or a target definition or create Siebel EIM transformations, the Designer connects to the Siebel application to import metadata.

The following figure shows Designer integration with Siebel:



The Designer uses the Java Data Bean API to perform the following functions:

- To retrieve metadata for Siebel business components when you import a source or a target definition.
- To retrieve metadata for EIM tables when you create the Siebel EIM Read or Load transformations.
- To retrieve metadata for the Siebel business service method when you create a Business Service transformation.

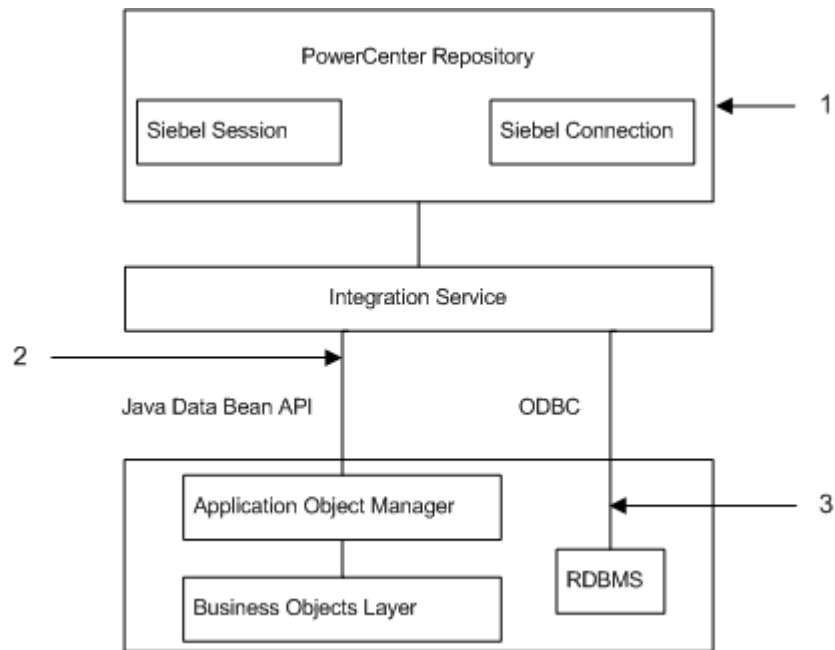
Note: The Designer does not connect to Siebel when you create the Siebel EIM Invoker transformation.

PowerCenter Integration Service and Siebel

The PowerCenter Integration Service connects to Business Objects Layer through the Java Data Bean API. The Java Data Bean API uses the Siebel Application Object Manager to access Siebel metadata in the Business Objects Layer. The PowerCenter Integration Service connects to an Application Object Manager to extract data from Siebel sources and load data into Siebel targets, or to invoke business services. The Application Object Manager handles connection requests to the Siebel server.

The PowerCenter Integration Service connects to the Relational Database Management System to extract or load bulk data into Siebel EIM tables through ODBC. The PowerCenter Integration Service connects to an Application Object Manager through Java Data Bean API to invoke an EIM task.

The following figure shows how the PowerCenter Integration Service integrates with Siebel:



1. The PowerCenter Integration Service reads and writes data based on session and application connection configuration.
2. The PowerCenter Integration Service uses the Java Data Bean API for extracting data from or loading data into Siebel business components, invoking business service, and invoking ServerRequest business service.
3. The PowerCenter Integration Service uses ODBC for extracting bulk data from or loading bulk data into Siebel database.

CHAPTER 2

PowerExchange for Siebel Configuration

This chapter includes the following topics:

- [PowerExchange for Siebel Configuration Overview, 17](#)
- [Registering the Plug-in, 18](#)
- [Configuring Java Properties, 19](#)
- [Copying Siebel API Libraries, 19](#)
- [Adding a Filter to Specify the Siebel Repository, 19](#)
- [Troubleshooting Configuration, 20](#)

PowerExchange for Siebel Configuration Overview

PowerExchange for Siebel installs with the Informatica services. Before you use PowerExchange for Siebel, you must complete the configuration tasks.

Configuring PowerExchange for Siebel

To configure PowerExchange for Siebel, perform the following steps:

1. Create a registry entry for PowerExchange for Siebel on the client machine:
 - a. Access the following directory:
`<Informatica installation directory>\clients\PowerCenterClient\client\bin`
 - b. Run the `PWX_SIEBEL_64.reg` file to create the registry entry.
2. Register the PowerExchange for Siebel plug-in.
3. Configure Java properties.
4. Copy Siebel API libraries.

After you configure PowerExchange for Siebel, you can create connections to access Siebel. Create connection objects in Workflow Manager so that the PowerCenter Integration Service can connect to Siebel at run time.

Registering the Plug-in

After you create a registry entry for PowerExchange for Siebel on the client machine, register the plug-in with the repository. If you are upgrading from a previous version, update the plug-in registration when you register the plug-in.

To register the plug-in, the repository must be running in exclusive mode. Use the Administrator tool or the pmrep RegisterPlugin command line program to register the plug-in. If you do not have the correct privileges to register the plug-in, contact the user who manages the PowerCenter Repository Service.

The plug-in file is an .xml file that defines the functionality of the adapter. When you install the server component, the installer copies the plug-in file to the following directory:

```
<Informatica installation directory>/server/bin/plugin
```

The name of the plug-in file for PowerExchange for Siebel is pmsiebel.xml.

Registering the Plug-in from the Administrator Tool

Register a repository plug-in to add its functionality to the repository.

1. Run the PowerCenter Repository Service in exclusive mode.
2. In the **Navigators**, select the PowerCenter Repository Service to which you want to add the plug-in.
3. In the **Contents** panel, click the **Plug-ins** view.
4. In the **Actions** menu of the **Domain** tab, select **Register Plug-in**.
5. On the **Register Plug-in** page, click the **Browse** button to locate the plug-in file.
6. Enter your user name, password, and security domain.

The **Security Domain** field appears when the Informatica domain contains an LDAP security domain.

7. Click **OK**.

The PowerCenter Repository Service registers the plug-in with the repository. The results of the registration operation appear in the activity log.

8. Run the PowerCenter Repository Service in normal mode.

Registering the Plug-in from the Command Line Program

You can use the pmrep RegisterPlugin command to register the plug-in from the command line program.

1. Run the PowerCenter Repository Service in exclusive mode.
2. Run the pmrep Connect command to connect to the Repository Service using a user account with Administrator Repository privilege.

The RegisterPlugin command uses the following syntax:

```
pmrep connect -r <repository name> -d <domain_name> -n <domain user name> -x  
<domain_password>
```

3. Find pmsiebel.xml in the following directory:

```
<Informatica installation directory>\server\bin\Plugin
```

4. Run the pmrep RegisterPlugin command to update the repository.

The RegisterPlugin command uses the following syntax:

```
pmrep registerplugin -i <Informatica installation directory>\server\bin\Plugin  
\pmsiebel.xml -e -N
```

Configuring Java Properties

Configure Java properties in the Administrator tool for each PowerCenter Integration Service process that runs Siebel sessions.

The following table describes the properties you configure:

Property	Description
Java SDK Classpath	You can set the CLASSPATH to any JAR files that you need to run a session that requires java components. The PowerCenter Integration Service appends the values you set to the system CLASSPATH.
Java SDK Minimum Memory	Minimum amount of memory the Java SDK uses during a session. If the session fails due to a lack of memory, you can increase this value. Default is 32 MB. If you want to increase the value, specify the value followed by M. For example, specify 70M for 70 MB.
Java SDK Maximum Memory	Maximum amount of memory the Java SDK uses during a session. If the session fails due to a lack of memory, you can increase this value. Default is 64 MB. If you want to increase the value, specify the value followed by M. For example, specify 90M for 90 MB.

Copying Siebel API Libraries

Copy the Siebel API libraries from the machine where Siebel is installed to the PowerCenter Client machine and the machines where the PowerCenter Integration Service process runs.

1. Locate the `Siebel.jar` and `SiebelJI_enu.jar` files in the following directory:
`<Siebel Installation Directory>\siebsrvr\CLASSES`
2. Copy the files to the following directory on the PowerCenter Client machine:
`<Informatica installation directory>\clients\PowerCenterClient\client\bin\javali`
3. Copy the files to the following directory on the machine that run the PowerCenter Integration Service process:
`<Informatica installation directory>\server\bin\javali`

Adding a Filter to Specify the Siebel Repository

If multiple Siebel repositories are available, you can configure the `connection.properties` file to add a filter to specify the repository name in the **Import from Siebel** wizard.

Note: If only one Siebel repository is available, you need not add the filter to specify the repository name.

1. Create the `connection.properties` file in the following directory:
`<Informatica installation directory>\clients\PowerCenterClient\client\bin`
2. Add the following entry to the `connection.properties` file:
`RepositoryName=MyRepo`

3. Save the `connection.properties` file.
4. Restart the PowerCenter Designer and launch the import wizard.
The **Repository Name** field appears in the **Connect to Siebel** page.

When you import Siebel business components, use the **Repository Name** field to specify the name of the Siebel repository if multiple Siebel repositories are available.

Troubleshooting Configuration

The Designer fails to open the Import from Siebel wizard.

The Designer cannot create Java Virtual Machine (JVM) on the PowerCenter Client machine if the PowerCenter Client machine does not have enough allocated memory. When this occurs, the `SiebelEIMClientLogC.log` file displays the following error:

```
Unable to create JVM
```

Verify that the PowerCenter Client machine has at least 64 MB of allocated memory for JVM. If you increase the allocated memory and get the same error, set the allocated memory to a higher value.

Complete the following steps to configure the allocated memory on the PowerCenter Client machine:

1. Create the `JVMOptions.ini` file in the following directory:
`<Informatica installation directory>\clients\PowerCenterClient\client\bin`
2. Enter the `Xmx` property in the `JVMOptions.ini` file and set it to 64, 128, 256, or 512 MB.

For example, you can enter:

```
Xmx=64
```

Note: The `Xmx` property sets the maximum allocated memory. If you get the same error after configuring the property, set it to a higher value.

CHAPTER 3

Siebel Sources

This chapter includes the following topics:

- [Siebel Sources Overview, 21](#)
- [Working with Siebel Sources, 21](#)
- [Generating Key Values, 23](#)
- [Filtering Siebel Sources, 25](#)
- [Siebel Source Definitions, 26](#)
- [Updating Siebel Source Definitions, 28](#)

Siebel Sources Overview

Siebel source definitions represent Siebel metadata. You can import Siebel source definitions through the Java Data Bean API. When you import a Siebel source definition through the Java Data Bean API, the Designer extracts the metadata from the business components on the Business Objects Layer.

Use the Import from Siebel Wizard to import Siebel business components. When you import a Siebel source definition, the Designer displays business objects for the active application object manager. You can select a business object and then a business component to import from the list of business components.

Working with Siebel Sources

A Siebel business component can contain single-value and multi-value fields. A single-value field can store one value at a time and a multi-value field can store more than one value at a time. A multi-value field has an associated multi-value link attribute that corresponds to the destination business component for this field.

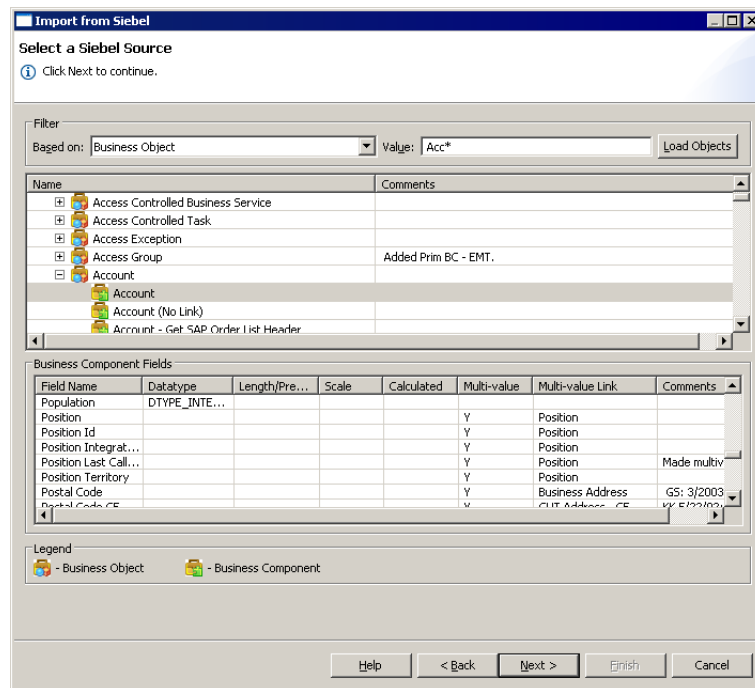
A multi-group source contains groups for single-value and multi-value fields. All the single-value fields are part of the first group. The multi-value fields with the same multi-value link are part of the same group. The group containing the single-value fields is the parent group for all the groups in that source. The other groups are child groups of the parent group.

The following table shows an example of the Account business component with fields and groups:

Fields	Field Name	Multi-value Link	Group	Destination Business Component
Single-value	Account	-	Account	-
Single-value	Account Number	-	Account	-
Multi-value	Street Address	Business Address	Business_Address	CUT Address
Multi-value	City	Business Address	Business_Address	CUT Address
Multi-value	Position	Position	Position	Position
Multi-value	Position ID	Position	Position	Position

An account name can be associated with multiple addresses. Therefore, each Account row can have multiple rows with Street Address and City columns. Multiple rows are stored as data of the CUT Address business component. An account name can be associated with multiple positions. Therefore, each Account row can have multiple rows with Position and Position ID columns. Multiple rows are stored as data of the Position business component.

The following figure shows the Account business component containing Position with a multi-value link attribute Position:



The Account source definition contains a group for Name and Account Number, a group for Position and Position ID, and the other group for City and Street Address.

Generating Key Values

When you import a multi-group Siebel source definition, the Designer creates a key relationship between each child group in the source definition and its parent group.

Note: When you create mappings with the primary key-foreign key relationship, the Designer might not create the relationship for the tables. You need to manually set the primary key relationship in the foreign key table.

The following table lists the primary and foreign keys created for the groups in the Account source definition:

Group	Primary Key	Foreign Key
Account	GPK__Account	-
Business_Address	GPK__Business_Address	GFK__Account__Business_Address
Position	GPK__Position	GFK__Account__Position

For a multi-group source, the PowerCenter Integration Service uses row ID as the group primary key of each group. The values for the group foreign keys are taken from the list of values for the group primary keys in the parent group.

The following table shows sample rows contained in the Account group:

GPK__Account	Name	Account_Number
1	James Brown	101
2	John Martin	102

The following table shows sample rows contained in the Business_Address child group:

GPK__Business_Address	GFK__Account__Business_Address	City	Street Address
A	1	Los Angeles	63 Ranch Ave
B	1	Frankfurt	3090 Stevens Creek Blvd
C	2	Dallas	181 Curtner Ave
D	2	San Diego	760 Camino Real

The following table shows sample rows contained in the Position child group:

GPK__Position	GFK__Account__Position	Position	Position ID
E	1	Account Manager	A1
F	1	Business Analyst	B1

GPK__Position	GFK__Account__Position	Position	Position ID
G	2	Consultant	C1
H	2	Financial Adviser	C2

In this example, each source represents a group in the source definition. To maintain key relationships when you connect mapping objects, connect source columns in the groups to the corresponding group in the flat file target definitions.

The following table lists column relationships between groups in the Account source definition and flat file target definitions:

Source Group Name	Source Column Name	Target Name	Target Column Name
Account	GPK__Account	Account_rdr1	GPK
Account	Name	Account_rdr1	Name
Account	Account_Number	Account_rdr1	Account_Number
Position	GPK__Position	Position_Account_rdr	GPK
Position	GFK__Account__Position	Position_Account_rdr	GFK
Position	Position	Position_Account_rdr	Position
Position	Position_ID	Position_Account_rdr	Position_ID
Business_Address	GPK__Business_Address	Business_Add_Account_rdr	GPK
Business_Address	GFK__Account__Business_Address	Business_Add_Account_rdr	GFK
Business_Address	City	Business_Add_Account_rdr	City
Business_Address	Street_Address	Business_Add_Account_rdr	Street_Address

The PowerCenter Integration Service extracts the following rows from the Account source definition:

```
[Account (1, "James Brown", "101", Address (A, 1, Los Angeles, 63 Ranch Ave), Address
(B, 1, Frankfurt, 3090 Stevens Creek Blvd), Position (E, 1, Account Manager, A1),
Position (F, 1, Business Analyst, B1)]
[Account (2, "John Martin", "102", Address (C, 2, Dallas, 181 Curtner Ave), Address (D,
2, San Diego, 760 Camino Real), Position (G, 2, Consultant, C1), Position (H, 2, Finance
Adviser, C2)]
```


The following table lists the rows that the PowerCenter Integration Service inserts into flat file targets:

Flat File Target Name	Inserted Rows
Account_rdr1	[1, James Brown, 101] [2, John Martin, 102]
Position_Account_rdr	[E, 1, Account Manager, A1] [F, 1, Business Analyst, B1] [G, 2, Consultant, C1] [H, 2, Finance Adviser, C2]
Business_Add_account_rdr	[A, 1, Los Angeles, 63 Ranch Ave] [B, 1, Frankfurt, 3090 Stevens Creek Blvd] [C, 2, Dallas, 181 Curtner Ave] [D, 2, San Diego, 760 Camino Real]

Filtering Siebel Sources

When you import Siebel source definitions, you can view business objects and business components that match the condition.

You can filter business objects and business components by name. Use the following guidelines when you enter a filter condition:

- Use AND or OR operators to enter more than one filter criterion.
- Use a question mark (?) as wildcard character for a single character and an asterisk (*) as wildcard character for multiple characters.
- If the filter condition contains special characters, enclose the filter condition in single quotes. Special characters include =, >, <, (,), ,, ~, ", ', and [.
- If the filter condition contains a single double quote or single quote, the quote must be doubled. For example, to search for a business component, FINS AG Agent's Designation or FINS AG Agent's Name, enter the following filter condition:

```
'FINS*'*'
```

- Enter date values in the mm/dd/yyyy format.

If you apply a filter on business components without selecting a business object, the wizard shows the business objects containing the business components that match the filter criterion. If you apply a filter on business components after selecting a business object, the wizard shows the business components that match the filter criterion in that business object.

After retrieving the names of business objects, select the business object for which you want to view the business components. To filter business components, select Business Component from the Based On list, enter the filter criteria in the Value, and click Load Objects.

To change the filter condition, enter the modified condition, and click Load Objects.

Siebel Source Definitions

When you connect to Siebel, the wizard lists a hierarchy of business objects and business components.

When you import a Siebel source definition, you import fields of a Siebel business component. In addition to importing predefined active fields of a Siebel business component, you can import the following types of special fields:

- **System field.** A system field is a standard Siebel field for a business component that represents the data from the system columns.
- **Custom field.** A custom field is a field that you can create to extract data from a field that exists in the Siebel business component.
- **Calculated field.** A calculated field is a Siebel field that derives its value from a calculated expression. You import Siebel calculated fields as custom fields.
- **Formatted field.** A formatted field is a field that you can create and use to extract data in the configured format. When you import a Siebel business component as a source definition, you can select the formatted fields and the PowerCenter Integration Service can read data of the formatted fields in the configured format. When you configure any field to retrieve data in a specific format, the datatype for the field changes to DTYPE_TEXT in the Siebel source definition.

The following table describes the details imported by the Designer when you import a business component as a source definition:

Source Definition Field	Description
Source Table	Business component name.
Column Name	The Designer displays the following attributes for each field: <ul style="list-style-type: none">- Field Name- Datatype- Length/Precision- Scale- Calculated- Multi-value- Multi-value link- Comments

Importing a Siebel Business Component

1. In the Source Analyzer, click Sources > Import Siebel Business Components.
The wizard appears.
2. Enter the connection parameters.

The following table describes the connection parameters:

Connection Parameter	Description
Protocol	Protocol used to connect to Siebel. The format for the protocol is siebel[.[transport][.[encryption][.[compression]]]]. The parameters in the protocol format are: <ul style="list-style-type: none">- transport. Enter http or TCPIP. Default is TCPIP.- encryption. Enter NONE or RSA. Default is NONE.- compression. Enter NONE or zlib. Default is zlib.
Siebel Server Host	Siebel server host name.
SCBroker Port	Siebel Connection Broker port number.
Enterprise Server	Siebel Enterprise Server name.
Application Object Manager	Siebel application business object manager.
Repository Name	Name of the Siebel repository. If multiple Siebel repositories exist, the Repository Name connection parameter appears if you have configured the connection.properties file to display this parameter.
User Name	Siebel user name to connect to Siebel.
Password	Password for user.
Encoding	Encoding used for communication between the Designer and the Siebel server. By default, the Designer uses the UTF-8 encoding.

For more information about the connection parameters, contact your Siebel system administrator.

3. Click Connect to connect to Siebel.
4. If the connection is successful, click Next.
5. Click Load Objects if you want to load all the business objects for the active application object manager.
6. Optionally, enter a filter condition to view business objects and business components that meet the condition.
7. Select a business object and drill down through the selected business object to select the business component you want to import. After you select a business component, the details area displays information about the business component.

Note: You can also import custom business components.

8. To add special fields, click Add Fields.

You can add the following types of special fields:

- System fields. In the System Fields tab, select the system fields, and then select Apply > OK.
- Custom or calculated fields. In the System Fields tab, click Add, and then enter the required parameters.

The following table describes the parameters used in the System Fields tab:

Field	Description
Name	Custom field name. The field name must begin with a character.
Datatype	Datatype for the field. Default is DTYPE_TEXT.
Length/Precision	Precision for the field. The precision must be between 1 and 104,857,600. Default value is the default precision of the selected Siebel datatype.
Scale	Scale for the field. The scale must be between 0 and 28. Default is 0.
Calculated	Indicates if the field is configured as a calculated field. Default is No.
Business Name	Business name for the field.
Destination Field	Destination field for the Siebel field.
Comments	Description for the field.

Select the custom field, and then click Apply > OK.

- Formatted field. In the Formatted Fields tab, select the fields, and then click Apply > OK.

9. Click Next.
10. Enter a name for the source in the Source Name field if you want to change the name of the source.
11. Click Import.

The wizard shows the fields and number of groups that the Designer created, and the number of groups and fields that the Designer failed to create.

Note: The Designer replaces special characters in the group and field names with an underscore (_). Special characters include space, ., / " + = ~ ` ! % ^ () [] { } ; : ; ? < > & @ * and |.

12. Click Import Another Source to import another source.
13. Repeat steps 1 to 12 to import another source definition.
14. Click Finish.

Updating Siebel Source Definitions

Manually edit the definition to configure the properties that you cannot import or to make minor changes to the definition.

You can update a Siebel source definition to change the name, datatype, precision, key type, and business name for a column.

Note: If the changes are significant, you can reimport the definition. This overwrites or renames the existing source definition.

Reimporting a Siebel Source Definition

When you reimport a Siebel source, you can retain the following information in the definition being replaced:

- Primary key-foreign key relationships
- Source definition description
- Column or port description

Editing a Siebel Source Definition

On the Columns tab, edit the precision and key type for fields in the source definition.

CHAPTER 4

Application Source Qualifier for Siebel Sources

This chapter includes the following topics:

- [Application Source Qualifier for Siebel Sources Overview, 30](#)
- [Source Filter, 30](#)
- [Sort Filter, 31](#)
- [View Mode, 32](#)

Application Source Qualifier for Siebel Sources Overview

In the Application Source Qualifier for the sources that are imported through the Java Data Bean API, you can perform the following tasks:

- Filter data. Use the source filter to retrieve the rows from a business component that satisfy a criterion.
- Sort data. Use the sort filter to sort data and to ensure that the PowerCenter Integration Service reads source data in a specific order from a business component.
- View data through view mode. View mode defines the visibility type for a business component.

Source Filter

Use the source filter to retrieve rows from a business component that meet a condition. Use the following guidelines when you enter a filter condition:

- Use AND or OR operators to enter more than one filter criterion.
- Use a question mark (?) as wildcard character for a single character and an asterisk (*) as wildcard character for multiple characters.
- If the filter condition contains special characters, enclose the filter condition in single quotes. Special characters include = > < () , ~ " ' and [.
- If the filter condition contains a single double quote or a single quote, you must double the quotes.

- Enter date values in the mm/dd/yyyy format.

You can also enter a filter condition that retrieves data from a parent group and a child group. For example, you can retrieve the accounts for which status is Active from the Account business component. You can also retrieve addresses of the Active accounts which have Country as U.S. from the Business_Address group. The Business_Address group is the child group of the Account group.

To compare a single-value field with multiple values, specify these values in a mapping parameter file. The mapping parameter file contains a mapping parameter. For example, to retrieve data from the Business_Address group about the persons who stay either in U.S., U.K., or India, declare the mapping variable in the Mapping Designer. Enter the name of the mapping variable preceded by \$\$ in the filter condition.

Note: Use the syntax provided by the Java Data Bean API to enter a filter condition.

1. In the Mapping Designer, double-click the Application Source Qualifier.
2. Click the Siebel Properties tab.
3. Click the Browse button by the Source Filter field.

The Search Builder dialog box appears.

4. Select the group to be filtered.
5. Select a field from the Field list.

Multi-value field names in the Field list are the same as in the destination business component. To view the corresponding field names in the source definition, double-click the source definition in the Source Analyzer, and click the Attributes tab. The Value column shows the field names as present in source definition.

6. Select the operator from the Operator list.
7. Enter the value for the selected field in Value.
Enter date values in the mm/dd/yyyy format.
8. Optionally, enter the mapping variable name in the Value field to compare the selected field with multiple values. The values are stored in the mapping parameter file.

Provide the name and path for the mapping parameter file at the session level.

9. Click Add to Filter to add the filter condition to the filter expression.
10. For multiple filter conditions, follow steps [1](#) to [9](#), and click AND into Filter or OR into Filter.
To add parentheses to a complex filter expression, click the Add buttons.

11. Click OK.

The filter condition appears in the Application Source Qualifier.

Sort Filter

Use the sort filter to retrieve data from a business component in a specific order. You can specify the sort condition for each group in a multi-group source. For example, you can retrieve data from the Business_Address group sorted by the City field. City is a multi-value field in the Account business component. Therefore, a row in the Account business component can have corresponding multiple rows in the CUT Address business component. The sort filter sorts the cities for a row in the Account business component and then across the rows in CUT Address.

1. In the Mapping Designer, double-click the Application Source Qualifier.

2. Click the Siebel Properties tab.
3. Click the Browse button by the Sort Filter field.
The Sort Builder dialog box appears.
4. Select the group to be sorted.
The Field list displays all the fields that correspond to the multi-value fields in the multi-value source for the selected group.
5. Select a field by which you want to sort the data.
Multi-value field names in the Field list are the same as in the destination business component. To view the corresponding field names in the source definition, double-click the source definition in the Source Analyzer, and click the Attributes tab. The Value column shows the field names as present in source definition.
6. Select the sorting order from the Type list.
7. Click Add to Filter to add the sort condition to the sort expression.
8. Click OK.
The sort condition appears in the Sort Filter in the Application Source Qualifier.

View Mode

The view mode determines the number of rows extracted by the PowerCenter Integration Service from a business component. You can specify the following values for view mode:

- SalesRep. Displays rows according to the following access control mechanisms used by a business component:
 - Single position. You can associate a single position with individual data. The PowerCenter Integration Service extracts the rows that are owned by the user position.
 - Sales team. You can associate multiple positions, in the form of a team, with data. The PowerCenter Integration Service extracts the rows that are owned by the users whose sales team contains the user position.

To view the rows for this view mode, the business component must have a view mode with an owner type of Position.
- Manager. Displays rows for a user and the rows accessible by the users who report to the user. The rows displayed depend on the access control mechanism. For the single position access control, the PowerCenter Integration Service extracts the rows associated directly with the active position of the user and the rows associated with subordinate positions.

For the sales team access control, the PowerCenter Integration Service extracts the row for which the active position of the user is the primary position on the team or a subordinate position is the primary member on the team.

The PowerCenter Integration Service does not display any data if a user position has no subordinate positions. To view the rows for this view mode, the business component must have a view mode with an owner type of Position. This view mode is used by managers.
- Personal. Displays the rows to which you have direct access. To view the rows, the business component must have a view mode with an owner type of Person. For example, you can view the rows for which you have permissions. This is the default view mode.

- All. Displays the rows for an organization that have a valid owner. For example, an administrator can view all the accounts in the Account business component.
- Organization. Displays rows for the organization to which a valid owner has been assigned and the position of the user is associated with the organization. This view mode is applicable to the single organization or multiple organization access control. For example, an executive can view the accounts of all the organizations in the Account business component.
- Group. Displays rows either in the current category or in the first level child categories of a category to which the user has access. The PowerCenter Integration Service extracts the rows in the first level category if the user accesses the catalog.
- Category. Displays a flat list of rows in categories across every catalog that a user can access.
- SubOrganization. Displays rows depending on the type of access control. For the single organization access control, the PowerCenter Integration Service extracts the rows associated directly with the active position of the user or the descendant organization of the user. For the multiple organization access control, the PowerCenter Integration Service extracts the rows for which the active organization of the user or a descendant organization is the primary organization.

When a Siebel user selects a view mode and the rows are not available for the selected view mode, the PowerCenter Integration Service writes a message to the session log for the view mode.

The following table lists the hierarchy in a sample organization:

User Name	Designation	Manager	Subordinates	Rows created for the Service Request Business Component
Emma	VP - Service	None	Alan, Shane	2
Alan	Service Manager	Emma	Shane	2
Shane	Service Representative	Alan	None	2

In this hierarchy, Emma, Alan, and Shane are Siebel users in the organization. Alan reports to Emma, and Shane reports to Alan. Every Siebel user has created two rows for the Service Request business component.

When a Siebel user uses the Personal view mode to read data from the Service Request business component, two rows are extracted that are created by the Siebel user.

When a Siebel user uses the Manager view mode for reading data, the data for the Siebel user and subordinates is extracted. For example, if Emma uses the Manager view mode, the rows created by Emma, Alan, and Shane are extracted. If Alan uses the Manager view mode, the rows created by Alan and Shane are extracted.

When a Siebel user uses the Organization view mode, the rows created by the Siebel user and the other Siebel users in the organization are extracted because all Siebel users belong to the same organization. For example, if Shane uses the Organization view mode, the rows created by Shane, Alan, and Emma are extracted.

CHAPTER 5

Siebel Targets

This chapter includes the following topics:

- [Siebel Targets Overview, 34](#)
- [Working with Siebel Targets, 34](#)
- [Generating Key Values, 36](#)
- [Configuring the Update Strategy, 36](#)
- [Siebel Target Definitions, 38](#)
- [Updating Siebel Target Definitions, 39](#)
- [Troubleshooting Siebel Targets, 40](#)

Siebel Targets Overview

Siebel target definitions represent the metadata for a Siebel business component. Use the Import from Siebel Wizard to import the Siebel target definitions. When you connect to Siebel, the wizard lists Siebel business objects. You can expand a business object to list the business components it contains.

When you select a business component to import, the details area of the wizard displays information such as the field names, datatypes, precision, and scale. The details area does not display the details for business objects.

When you import a business component that contains multi-value links corresponding to the fields from other business components, the Designer imports all the fields as one target definition.

Working with Siebel Targets

A Siebel business component can contain single-value and multi-value fields. A single-value field can store one value at a time and a multi-value field can store more than one value at a time. A multi-value field has an associated multi-value link attribute that corresponds to the destination Siebel business component for this field.

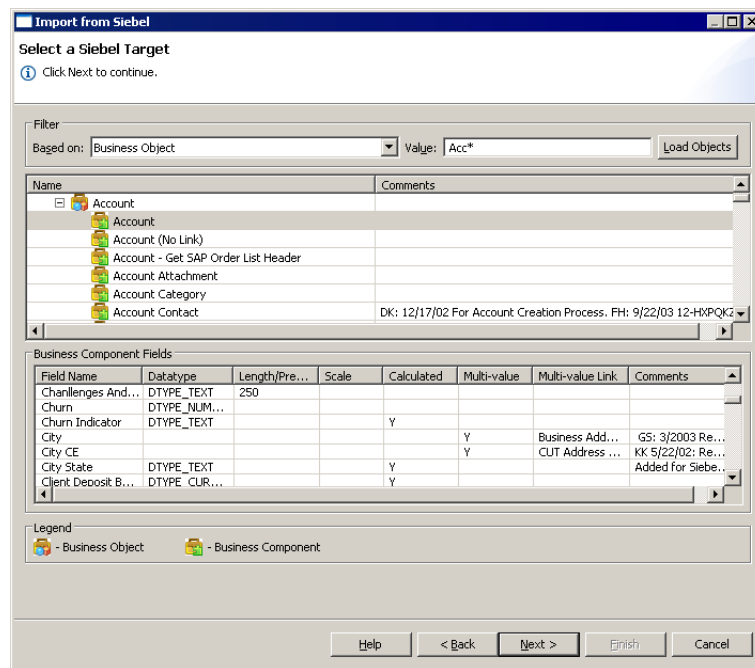
A multi-value target contains groups for single-value and multi-value fields. All the single-value fields are part of a single group. The multi-value fields with the same multi-value link are part of the same group. The group containing the single-value fields is the parent group for all the groups in that target. The other groups are child groups of the parent group.

The following table shows an example of the Account business component with fields and groups:

Fields	Name	Multi-Value Link	Group	Destination Business Component
Single-Value	Name	-	Account	-
Single-Value	ID	-	Account	-
Multi-Value	Street Address	Business Address	Business_Address	CUT Address
Multi-Value	City	Business Address	Business_Address	CUT Address

Each Account row can have multiple rows with Street Address and City columns. A name can be associated with multiple addresses. Multiple rows are stored as data of the CUT Address business component.

The following figure shows the Account business component containing City with a multi-value link attribute, Business Address:



Account contains a group for Name and ID and the other group for City and Street Address.

When you import a Siebel business component as a target definition, the Designer imports the following business component properties:

- Business component name. Displays the name of the business component.
- Business object name. Displays the business object name for the business component.
- Field. Displays the name of the field.

Generating Key Values

When you import a Siebel target definition, the Designer creates a key relationship between each group in the target definition and its parent group. Each key uses the following naming convention:

GPK__<group_name>

GFK__<parentgroup_name>__<group_name>

The following table describes the key naming conventions:

Key Name Component	Description
GPK GFK	Type of key. The key name begins with GPK when it is a primary key. The key name begins with GFK when it is a foreign key.
group_name	Name of the group to which the key belongs.
parentgroup_name	Name of the parent group with which you have established primary key relationship.

For the Account business component, the Designer generates primary keys, GPK__Account and GPK__Business_Address. It generates the GFK__Account__Business_Address foreign key for the child group.

Provide the values for the GPK and GFK fields in the source that define the relationship between the parent group and child group rows. Choose values for the GFK field from the GPK field in the parent group.

Note: When you create mappings with the primary key-foreign key relationship, the Designer might not create the relationship for the tables. You need to manually set the primary key relationship in the foreign key table.

Configuring the Update Strategy

Each group in the target business component contains a field that determines the update strategy for the group. Configure this field in the mapping to specify the operation that you need to perform on the rows. You can perform INSERT, UPDATE, and DELETE operations on rows. The default operation is INSERT.

The following table lists the valid values for the update strategy field:

Operation	Value
INSERT	0
UPDATE AS UPDATE	1
DELETE	2
UPDATE ELSE INSERT	3

If you specify an invalid value for this field, the PowerCenter Integration Service treats it as INSERT.

Note: The PowerCenter Integration Service ignores the value of the Treat Source Rows As property at the session level.

The naming convention for the update strategy field is Update_Strategy_<group_name>. The Account group contains the Update_Strategy_Account field and the Business_Address group contains the Update_Strategy_Business_Address field.

The following table shows the rows contained in the source definition Account_Src:

GPK__Account	Update_Strategy_Account	Name	ID
1	0	James Brown	101
2	0	John Martin	102

The following table shows the rows contained in the source definition Business_Add_Src:

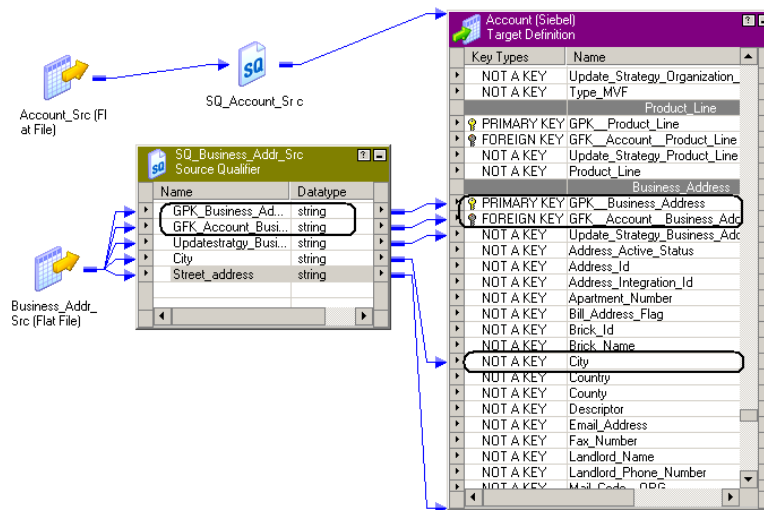
GPK__Business_Address	GFK__Account_Business_Address	Update_Strategy_Business_Address	City	Street Address
A	1	0	Los Angeles	63 Ranch Ave
B	1	0	Frankfort	3090 Stevens Creek Blvd
C	2	0	Dallas	181 Curtner Ave
D	2	0	San Diego	760 Camino Real

In this example, each source represents a group in the target definition. To maintain key relationships when you connect mapping objects, connect source columns to the corresponding group in the target definition.

The following table lists the column relationships between the source definitions and the groups in the Account target definition:

Source Name	Source Column Name	Account Target Group Name	Target Column Name
Account_Src	GPK__Account	Account	GPK__Account
Account_Src	Name	Account	Name
Account_Src	ID	Account	ID
Business_Addr_Src	GPK__Business_Address	Business_Address	GPK__Business_Address
Business_Addr_Src	GFK__Account__Business_Address	Business_Address	GFK__Account__Business_Address
Business_Addr_Src	City	Business_Address	City
Business_Addr_Src	Street_Address	Business_Address	Street_Address

The following figure shows the Account business component with a multi-valued field:



The PowerCenter Integration Service inserts the following rows into the target business component:

```
[Account ("James Brown", "101", Address (Los Angeles, 63 Ranch Ave), Address (Frankfort,
3090 Stevens Creek Blvd)]
[Account ("John Martin", "102", Address (Dallas, 181 Curtner Ave), Address (San Diego,
760 Camino Real)]
```

Siebel Target Definitions

When you connect to Siebel, the wizard lists a hierarchy of business objects and business components.

Importing a Siebel Target Definition

1. In the Target Designer, click Target > Import from Siebel.

The wizard appears.

2. Enter the connection parameters.

The following table describes the connection parameters:

Connection Parameter	Description
Protocol	<p>Protocol used to connect to Siebel. The format for the protocol is siebel[.[transport]][.[encryption]][.[compression]]].</p> <p>The parameters in the protocol format are:</p> <ul style="list-style-type: none"> - transport. Enter http or TCPIP. Default is TCPIP. - encryption. Enter NONE or RSA. Default is NONE. - compression. Enter NONE or zlib. Default is zlib.
Siebel Server Host	Siebel server host name.

Connection Parameter	Description
SCBroker Port	Siebel Connection Broker port number.
Enterprise Server	Siebel Enterprise Server name.
Application Object Manager	Siebel Application Business Object Manager.
Repository Name	Name of the Siebel repository. If multiple Siebel repositories exist, the Repository Name connection parameter appears if you have configured the connection.properties file to display this parameter.
User Name	Siebel user name to connect to Siebel.
Password	Password for user.
Encoding	Encoding used for communication between the Designer and the Siebel Server. By default, the Designer uses the UTF-8 encoding.

For more information about the connection parameters, contact your Siebel system administrator.

3. Click Connect to connect to Siebel.
4. If the connection is successful, click Next.
5. Click Load Objects if you want to load all the business objects for the active application object manager.
6. Optionally, enter a filter to view business objects and business components that meet the condition.
7. Select a business object and drill down to select the business component you want to import. After you select a business component, the details area displays the information about the business component.

Note: You can also import custom business components. When you import a Siebel business component as a target definition, the Designer imports all the fields except the calculated and read-only fields.

8. Click Next.
9. Enter a name for the source in the Target Name field if you want to change the name of the target.
10. Click Import.

The wizard shows the fields and number of groups that the Designer created, and the number of groups and fields that the Designer failed to create.

Note: The Designer replaces special characters in the group and field names with an underscore (_). Special characters include space, ., ,, /, ", +, =, ~, ` , !, %, ^, (,), [,] , { , } , : , ; , ? , < , > , & , @ , * , and |.

11. Click Import Another Target to import another target.
12. Repeat steps [1](#) to [10](#) to import another target definition.
13. Click Finish.

Updating Siebel Target Definitions

Manually edit the definition to configure properties that you cannot import or to make minor changes to the definition.

You can update a Siebel target definition to configure column attributes such as precision and key type.

Note: If the changes are significant, you can reimport the definition. This overwrites or renames the existing target definition. You can retain existing primary key-foreign key relationships and descriptions in the target definition being replaced.

Reimporting a Siebel Target Definition

When you reimport a Siebel target, you can retain the following information in the definition being replaced:

- Primary key-foreign key relationships
- Target definition description
- Column or port description

Editing a Siebel Target Definition

On the Columns tab, edit the precision and key type for fields in the business component. When you set the key type of columns as PRIMARY KEY, the PowerCenter Integration Service uses these columns to search the rows in the Siebel business components. You cannot modify the datatype of a column.

Troubleshooting Siebel Targets

The Import from Siebel option is not enabled in the Designer.

Verify that the Siebel license is assigned to the PowerCenter Repository Service.

CHAPTER 6

Siebel Business Service Transformation

This chapter includes the following topics:

- [Siebel Business Service Transformation Overview, 41](#)
- [Siebel Business Service Transformation Components, 41](#)
- [Siebel Business Service Transformation Groups and Ports, 42](#)
- [Using a Siebel Business Service Transformation in a Mapping, 42](#)
- [Creating a Siebel Business Service Transformation, 45](#)

Siebel Business Service Transformation Overview

A Siebel Business Service transformation is a passive transformation that you can use to invoke business service methods. Use Siebel business services to move data and convert data formats between Siebel and external applications. For example, the Authentication business service is a Siebel business service that takes credentials of users as the input and generates the user name and role of the user as output.

Business service methods transfer data between Siebel and external applications. You can use a Siebel Business Service transformation to perform complex calculations or specialized functions.

When you run a session with a Siebel Business Service transformation, the PowerCenter Integration Service connects to a Siebel server and invokes the business service method.

Use the Import from Siebel Wizard to create a Siebel Business Service transformation. When you connect to Siebel, the wizard displays the metadata for the business services. In the wizard, select a Siebel business service method to generate Siebel Business Service transformation. In addition to the primitive datatypes such as Number, Date, and String, business methods also take integration objects and hierarchy objects as arguments.

Siebel Business Service Transformation Components

A Siebel Business Service transformation contains the following tabs:

- Transformation. Enter a name and description for the Siebel Business Service transformation.
- Ports. View and edit port information on the Ports tab.

- **Properties.** View the properties on the Properties tab. Configure the tracing level property for the Siebel Business Service transformation.
- **Metadata Extensions.** The Designer provides vendor-defined metadata extensions that extend metadata stored in the repository. You cannot create user-defined metadata extensions.
- **Initialization Properties.** The Siebel Business Service transformation does not use initialization properties. The PowerCenter Integration Service retrieves initialization information from a vendor-defined metadata extension.
- **Port Attribute Definitions.** The Siebel Business Service transformation does not use port attributes.

Metadata Extensions

A Siebel Business Service transformation provides the following vendor-defined metadata extensions:

- **BusinessServiceMethodName.** Refers to the name of the business service method represented by the transformation.
- **BusinessServiceName.** Refers to the name of the business service to which the business service method belongs.

The Designer sets the values of metadata extensions. You cannot edit these values.

Siebel Business Service Transformation Groups and Ports

A Siebel Business Service transformation has input and output groups. The naming convention for the input group is <Business Service name>_In. The naming convention for the output group is <Business Service name>_Out. Siebel Business Service transformation can have multiple input and output ports. The input arguments of a business service method correspond to the input ports, and the output arguments correspond to the output ports.

Note: If a business service method does not have any input or output port, a dummy port is added to the input or output group.

Using a Siebel Business Service Transformation in a Mapping

A Siebel Business Service transformation can have the following types of input and output arguments:

- **Integration object.** Provides a logical representation of external application data or Siebel data that corresponds to an external application.
- **Hierarchy object.** Represents hierarchical Siebel data stored in the Siebel application.
- **Date, String, or Number.** Date, String, and Number are primitive datatypes.

The following table describes the input and output ports of a Siebel Business Service transformation:

Input Argument	Output Argument	Description
Integration object	Date, String, or Number	An integration object is passed as an XML string to a Siebel Business Service transformation. You can directly pass the XML string to the Siebel Business Service transformation or use an XML Generator transformation to generate the integration object as an XML string.
Date, String, or Number	Integration object	An integration object is generated as an XML string by the Siebel Business Service transformation. Use an XML Parser transformation to parse the XML string generated by the Siebel Business Service transformation.
Integration object	Integration object	You can directly pass the XML string to the Siebel Business Service transformation or use an XML Generator transformation to generate the integration object as an XML string. Use an XML Parser transformation to parse the XML string generated by Siebel Business Service transformation.
Hierarchy object	Date, String, or Number	A hierarchy object is directly passed as an XML string to a Siebel Business Service transformation.
Date, String, or Number	Hierarchy object	A hierarchy object is generated as an XML string by a Siebel Business Service transformation.
Hierarchy object	Hierarchy object	A hierarchy object is directly passed as an XML string to a Siebel Business Service transformation. A hierarchy object is generated as an XML string by the Siebel Business Service transformation.

Working with Integration Objects

When a Siebel Business Service transformation invokes a method that takes an integration object as an argument, you can use an XML Generator transformation to generate the XML string.

Each integration object has an XSD generated by the Designer. The XSD is stored in the <PowerCenter Installation Directory>\clients\PowerCenterClient\client\bin directory. The format for the XSD file name is <IntegrationObjectName>.xsd. The data you provide as input to a Siebel Business transformation must comply with the XSD. The XML Generator transformation converts the source data to a format compliant with the XSD. The output of the XML Generator transformation is input for the Siebel Business Service transformation.

When a business service method returns an integration object as an output, you can parse the output XML according to the target database using an XML Parser transformation. The Designer generates XSD for the integration object. The XML Parser transformation consumes the XSD and parses the output XML according to the target database. For example, if you want to insert data into an Oracle table, you can parse the XML output to a format that complies with the Oracle table.

Example of Integration Object as Input

A business service method can take an integration object as the input. Consider a business case where you want to load data from a flat file source into a Quote business component. You can use the Siebel Quote business service to invoke a method called Insert. The Insert method takes the integration object SiebelMessage as an input argument and adds a record to the underlying business component called Quote. The addition of a record is based on the input SiebelMessage, which is an XML string. To generate the XML string, use the XML Generator transformation.

The following figure shows a mapping that includes a Siebel Business Service transformation:

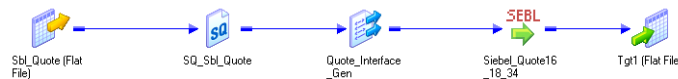


To insert a Quote record, provide input values for Name, Id, QuoteNumber, QuoteType, and Description from the source. The XML Generator transformation uses the input values to create an integration object in the XML format. Use Quote Interface.xsd to create the XML Generator transformation that generates the XML string in the format required by the Insert method. Pass the output of the XML Generator transformation to the Siebel_Quote_Insert Business Service transformation.

Example of Integration Object as Output

A business service method can also generate an integration object as an output. Consider a case where you want to query the details of an account. You can use the Siebel Account business service to invoke a method called QuerybyId. QuerybyId accepts the primaryRowId of an account as an input and generates the integration object in an XML string format as the output. You can use an XML Parser transformation to parse the output XML string into a format compliant with the target.

The following figure shows a Siebel Business Service transformation that invokes the business service method called QuerybyId:



To get the account details, provide the primaryRowId of an account from the source. The Siebel_Account_QueryById transformation passes the account details as an integration object, in an XML string format, to the SiebelMessage port. Use the Account Interface.xsd to create an XML Parser transformation. Pass the output of the Siebel_Account_QueryById transformation to the XML Parser transformation, which parses the XML string in the format required by the target.

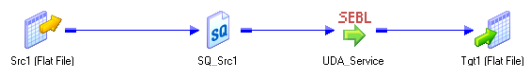
Working with Hierarchy Objects

A business service method can take a hierarchy object as the input. A business service method can also generate a hierarchy object as an output. The hierarchy object is represented as an XML string.

Example of Hierarchy Object as Input

You need to generate a hierarchy object as an XML string to pass it as an input argument to a Siebel business service method.

The following figure shows a sample mapping that uses a hierarchy object as the input:

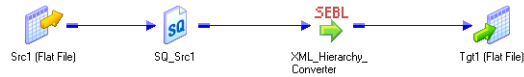


The sample mapping shows a Siebel Business Service transformation that takes a hierarchy object as the input and converts it into an XML document. The hierarchy object is passed as an XML string from the source to the Siebel Business Service transformation. This transformation converts the input hierarchy object into an XML document and generates the document as an XML string.

Example of Hierarchy Object as Output

A Siebel Business Service transformation generates a hierarchy object as an XML string.

The following figure shows a sample mapping that generates hierarchy object as an output:



The sample mapping shows a Siebel Business Service transformation that takes an XML document as an input and converts the input into the hierarchy object. The XML document is passed as an input to the Siebel Business Service transformation. This transformation converts the input XML document to the hierarchy object and generates the hierarchy object as an output.

Working with Strings, Numbers, and Dates

When a business service method takes a string, number, or date as an input argument, you can pass the input directly to a Siebel Business Service transformation.

For example, you can use a Siebel Business Service transformation, `Authentication_Business_Service`, to extract the role and Siebel user name of a Siebel user. This Siebel Business Service transformation accepts the user name, credentials, and password as inputs. This transformation provides the role and Siebel user name of the Siebel user as the output.

The following table lists the column relationships between the source definition, `Authentication_Business_Service`, and the target definition:

Source Column Name	Business Service Transformation Column Name	Target Column Name
GetPrivateCredentials	GetPrivateCredentials	-
Password	Password	-
User_Name	User_Name	-
-	Role	Role
-	Siebel_User_Name	Siebel_User_Name

Creating a Siebel Business Service Transformation

You can create a Siebel Business Service transformation in the Transformation Developer or the Mapping Designer.

1. Open the Transformation Developer.
Or, open the Mapping Designer.
2. Click Transformation > Create.
3. Select Siebel Business Service transformation.
4. Enter a name for the transformation.

5. Click Create.

The wizard appears.

6. Enter the connection parameters.

The following table describes the connection parameters:

Connection Parameter	Description
Protocol	Protocol used to connect to Siebel. The format for the protocol is siebel[.[transport]][.[encryption]][.[compression]]]. The parameters in the protocol format are: <ul style="list-style-type: none">- transport. Enter http or TCPIP. Default is TCPIP.- encryption. Enter NONE or RSA. Default is NONE.- compression. It can accept two values: NONE or zlib. Default is zlib.
Siebel Server Host	Siebel server host name.
SCBroker Port	Siebel Connection Broker port number.
Enterprise Server	Siebel Enterprise Server name.
Application Object Manager	Siebel Application Business Object Manager.
User Name	User name to connect to Siebel.
Password	Password for user.
Encoding	Code page used for communication between the Designer and the Siebel Server. By default, Java Data Bean API uses the UTF-8 encoding.

For more information about connection parameters, contact your Siebel system administrator.

7. Click Next.

The wizard displays a list of business services and the corresponding methods.

8. Select a business service to view the list of supported methods.

9. Select a method that you want to import.

When you select a method, you can view information about its input and output arguments in the details area.

10. Click Finish to import the selected method as a transformation.

11. Click Done.

CHAPTER 7

Siebel EIM Read Transformation

This chapter includes the following topics:

- [Siebel EIM Read Transformation Overview, 47](#)
- [Siebel EIM Read Transformation Components, 47](#)
- [Siebel EIM Read Transformation Groups and Ports, 49](#)
- [Using a Siebel EIM Read Transformation in a Mapping, 51](#)
- [Creating a Siebel EIM Read Transformation, 53](#)
- [Understanding the Read Summary for the Siebel EIM Read Transformation, 55](#)

Siebel EIM Read Transformation Overview

A Siebel EIM Read transformation is an active transformation that you can use to extract data from Siebel EIM tables. When you run a session with a Siebel EIM Read transformation, the PowerCenter Integration Service connects to a Siebel server and extracts data from Siebel EIM tables. The PowerCenter Integration Service writes the read summary in the session log for each group of a partition.

Siebel EIM Read Transformation Components

A Siebel EIM Read transformation contains the following tabs:

- Transformation. Enter the name and description of the transformation. The naming convention for a Siebel EIM Read transformation is `SRT_TransformationName`. You can also make the transformation reusable.
- Ports. View ports on the ports tab.
- Properties. View properties on the Properties tab. Configure the Tracing Level property for the Siebel EIM Read transformation.
- Read Transformation Properties. Configure the row status or filter to extract rows from Siebel EIM tables. If a transformation is a reusable transformation, you cannot edit the transformation in the Mapping Designer.

Note: You cannot modify the Ports tab in a Siebel EIM Read transformation.

Configuring Siebel EIM Read Transformation Properties

Configure transformation properties on the Properties tab and the Read Transformation Properties tab.

The following table describes the Siebel EIM Read transformation properties:

Property	Property Location	Description
Tracing Level	Properties tab	Amount of transaction detail reported in the session log file. Use the following tracing levels: <ul style="list-style-type: none">- Terse- Normal- Verbose Initialization- Verbose Data Default is Normal.
Rows to Read	Read Transformation Properties tab	Extracts rows from Siebel EIM tables based on the following options: <ul style="list-style-type: none">- Successful. Extracts the rows where the IF_ROW_STAT port is set to IMPORTED, EXPORTED, MERGED, or DELETED.- Erroneous. Extracts the rows where the IF_ROW_STAT port is set to a value other than EXPORTED, IMPORTED, MERGED, DELETED, FOR_IMPORT, FOR_MERGE, FOR_EXPORT, or FOR_DELETE.- All. Extracts all rows. Default is All.
Filter	Read Transformation Properties tab	Filters the rows in the Siebel EIM table. Use the following syntax to filter the rows: [<EIM table name>:<SQL WHERE clause>]

Extracting Rows from Siebel EIM Tables

The PowerCenter Integration Service extracts rows based on the following elements:

- Row status
- Filters
- Control fields

Extracting Rows Based on Statuses

You can extract rows from Siebel EIM tables based on the value of the IF_ROW_STAT column of the EIM tables. To extract the rows based on the statuses, configure the Rows to Read transformation property in the Read Transformation Properties tab of the Siebel EIM Read transformation.

Extracting Rows Based on a Filter

You can configure the filter in the Read Transformation Properties tab of the Siebel EIM Read transformation. Use the following syntax when creating the filter statement:

```
[<EIM table name>:<SQL WHERE clause>]
```

For example, the Siebel EIM Read transformation contains two input groups that correspond to the EIM tables EIM_AC_RSRC and EIM_ACCNT_CUT.

The following table shows the filtered queries you can apply:

EIM Table Name	SQL WHERE Clause
EIM_AC_RSRC EIM	select * from EIM_AC_RSRC where ACR_NAME like 'Acr%'
EIM_ACCNT_CUT EIM	select * from EIM_ACCNT_CUT where PARTY_UID='1-D1916' and AC_NAME like 'Acc%'

To apply the filters, enter the following filter value:

```
[EIM_AC_RSRC:ACR_NAME like 'Acr%'] [EIM_ACCNT_CUT:PARTY_UID='1-D1916' and AC_NAME like 'Acc%']
```

Note: If the sort order of a database is case-sensitive, specify the EIM table name in uppercase letters.

Siebel EIM Read Transformation Groups and Ports

A Siebel EIM Read transformation contains the following input and output groups:

- INPUT_STATUS input group
- CONTROL_FIELDS input group
- <Siebel EIM table name> output group
- OUTPUT_STATUS output group

The field name is stored in the Business Name attribute of the port. The description of the port contains more information about the port.

Note: You cannot edit the datatype, precision, or scale of any port.

INPUT_STATUS Input Group

The PowerCenter Integration Service extracts data from Siebel EIM tables based on the value of the INPUT_STATUS port in the INPUT_STATUS input group. The INPUT_STATUS port can contain the value 1 or 0. The PowerCenter Integration Service does not extract data from EIM tables if the value of this port is set to 0. If the value of the INPUT_STATUS port does not equal 0 or if this port is not linked, the PowerCenter Integration Service extracts data from EIM tables.

CONTROL_FIELDS Input Group

The PowerCenter Integration Service extracts data from EIM tables based on each unique combination of the EIM table name and the batch number specified in the CONTROL_FIELDS input group. The naming convention for each port is <group name>_<field name>. The CONTROL_FIELDS group contains the following ports:

- CONTROL_FIELDS_EIM_TABLE. Contains the EIM table name.
- CONTROL_FIELDS_BATCH_NUM. Contains the batch number.

The PowerCenter Integration Service retrieves the EIM table name and the batch number from a source that is linked to the CONTROL_FIELDS group. You can filter rows by specifying particular batch numbers and EIM tables in the control fields.

If a flat file provides values for the CONTROL_FIELDS group, the flat file header must contain the EIM table name and the batch number in the following format:

<Siebel EIM table name>,<batch number>

The PowerCenter Integration Service extracts data from an EIM table for each batch number specified in this group. If the specified batch number does not match a batch number in the EIM table, the PowerCenter Integration Service ignores that batch for session performance. If the source does not provide an EIM table name or batch number, the PowerCenter Integration Service does not extract data for any EIM table.

The following table shows an example EIM table, EIM_ACCNT_CUT:

ROW_ID	IF_ROW_BATCH_NUM
1-D1916	1
1-D1917	1
1-D1918	1
1-D1919	2
1-D1920	2
1-D1921	2

The following table describes the CONTROL_FIELDS group:

EIM Table Port Name	EIM Table Value	Batch Number Port Name	Batch Number Value
CONTROL_FIELDS_EIM_TABLE	EIM_ACCNT_CUT	CONTROL_FIELDS_BATCH_NUM	2

The PowerCenter Integration Service extracts data only for batch number 2 because the CONTROL_FIELDS group does not contain the batch number 1 for the EIM_ACCNT_CUT EIM table.

Rules and Guidelines for the CONTROL_FIELDS Input Group

Use the following rules and guidelines for the CONTROL_FIELDS input group of the Siebel EIM Read transformation:

1. The EIM table name must be in uppercase letters for the CONTROL_FIELDS_EIM_TABLE column.
2. Ensure that the number of rows in the CONTROL_FIELDS group does not exceed 5000 rows for better performance.
3. If you filtered the rows read from the source based on the batch number, verify that all ports in the CONTROL_FIELDS group are linked.

<Siebel EIM Table Name> Output Group

A Siebel EIM Read transformation contains one output group for each EIM table that is defined in the transformation. Each output group contains ports that correspond to the columns in the corresponding Siebel EIM table. The PowerCenter Integration Service extracts data from the linked columns of EIM tables and writes the extracted data into this group for each EIM table. The naming convention for each port is <group name>_<field name>.

OUTPUT_STATUS Output Group

When you use the Siebel EIM Read transformation to extract data from EIM tables, the PowerCenter Integration Service writes the output status to the OUTPUT_STATUS port in the OUTPUT_STATUS group. The OUTPUT_STATUS port can contain the values 1 or 0. If the PowerCenter Integration Service extracts the data successfully, the value of the OUTPUT_STATUS port is set to 1. Otherwise, the value of the OUTPUT_STATUS port is set to 0.

Using a Siebel EIM Read Transformation in a Mapping

You can use a Siebel EIM Read transformation in a mapping for the following scenarios:

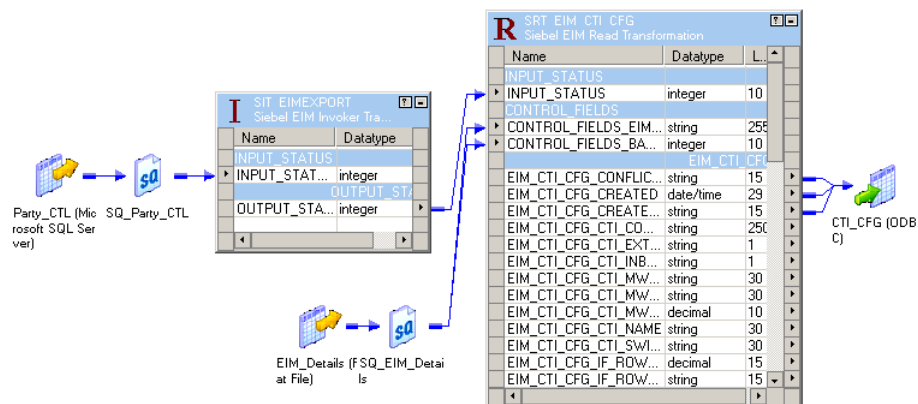
- Write Siebel base table data to a corporate data source.
- Determine the row-level status of Siebel EIM functions.

Write Siebel Base Table Data to a Corporate Data Source

You can extract data from Siebel base tables, and then load the data into a corporate data source. To load data from Siebel base tables to a corporate data source, perform the following steps:

1. Use a Siebel EIM Invoker transformation to invoke an EIM task that exports the data from the Siebel base table and write it to the Siebel EIM table.
2. Use a Siebel EIM Read transformation to extract rows from the EIM table such as successfully exported rows from Siebel base tables into Siebel EIM tables.

The following figure shows a Siebel EIM Read transformation mapping that extracts data from Siebel EIM tables and loads the data into a corporate data source:



The mapping contains the following components that are used to extract bulk data from Siebel EIM tables:

- **Party_CTL**. A source flat file that provides a value to the **INPUT_STATUS** port of the Siebel EIM Invoker transformation. When you run the session, the PowerCenter Integration Service invokes the EIM task if the value of the **INPUT_STATUS** port of the Siebel EIM Invoker transformation is not set to 0.
- **SQ_Party_CTL**. An Application Source Qualifier for the source instance that links the input status in the flat file to the **INPUT_STATUS** port of the Siebel EIM Invoker transformation.

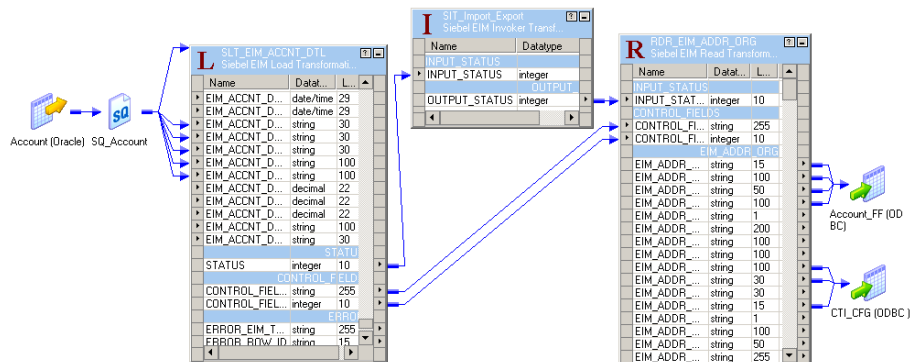
- **SIT_EIMExport.** A Siebel EIM Invoker transformation that invokes the Siebel EIM task specified in the IFB file and provides a value to the INPUT_STATUS port of the Siebel EIM Read transformation. Siebel EIM runs the EIM task to export the rows from the base table and writes these rows into the EIM tables. Siebel EIM writes the batch number in the IF_ROW_BATCH_NUM column and the status EXPORTED in the IF_ROW_STAT column of the EIM table for all successfully exported rows. After the PowerCenter Integration Service invokes the EIM task, it writes the output status to the OUTPUT_STATUS port of the Siebel EIM Invoker transformation.
- **EIM_Details.** A source flat file that provides a EIM table name and batch number to the CONTROL_FIELDS group of the Siebel EIM Read transformation.
- **SQ_EIM_Details.** An Application Source Qualifier that provides values to the CONTROL_FIELDS_EIM_TABLE and CONTROL_FIELDS_BATCH_NUM ports of the Siebel EIM Read transformation.
- **SRT_EIM_CTL_CFG.** A Siebel EIM Read transformation that extracts data from EIM tables. If the Siebel EIM Read transformation receives the value 1 for the INPUT_STATUS port, the PowerCenter Integration Service extracts all successfully exported rows for each EIM table and batch number specified in the CONTROL_FIELDS group. The PowerCenter Integration Service then writes the output status to the OUTPUT_STATUS port of the Siebel EIM Read transformation. It also writes the read summary to the session log.
- **CTL_CFG.** The PowerCenter Integration Service loads the data read from the EIM table into the CTL_CFG flat file target.

Determining the Row-Level Status of Siebel EIM Functions

You can use the Siebel EIM Read transformation in a mapping to determine the row-level statuses of the Siebel EIM functions. You can configure the session to only extract particular types of rows from EIM tables, such as erroneous rows. For example, you can use the Siebel EIM Read transformation in a mapping to extract data for the rows that are not successfully imported from Siebel EIM tables into Siebel base tables.

1. Use a Siebel EIM Load transformation to load data from a source to the Siebel EIM table.
2. Use a Siebel EIM Invoker transformation to invoke an EIM task that imports the data from the Siebel EIM table and write it to the Siebel base table.
3. Use a Siebel EIM Read transformation to extract rows based on a particular status, such as erroneous rows.

The following figure shows a mapping that extracts rows that were not successfully imported from Siebel EIM tables into Siebel base tables, and then loads the rows into corporate data sources:



The mapping contains the following components that load bulk data into Siebel base tables, and then exports rows that were not successfully loaded into the Siebel base table:

- **Account.** An Oracle source that provides rows to be loaded into the Siebel base table and the input status for the INPUT_STATUS port of the Siebel EIM Load transformation.
- **SQ_Account.** An Application Source Qualifier that links the source rows and input status to the input ports and INPUT_STATUS port of the Siebel EIM Load transformation.
- **SLT_EIM_ACCNT_DTL.** A Siebel EIM Load transformation that loads data from the source into the Siebel EIM tables and writes the batch number and EIM table name in the CONTROL_FIELDS output group for the rows loaded into EIM tables. It provides these values for the CONTROL_FIELDS group of the Siebel EIM Read transformation. The Siebel EIM Load transformation also outputs the status of the load to the Siebel EIM Invoker transformation.
- **SIT_Import_Export.** A Siebel EIM Invoker transformation invokes the EIM task to load the Siebel base table. Siebel EIM runs the EIM task on the Siebel server to import the rows from the EIM tables into the base tables specified in the IFB file. It writes the batch number in the IF_ROW_BATCH_NUM column and updates the statuses in the IF_ROW_STAT column of the EIM table for all the rows marked for import. After it invokes the EIM task, the Siebel EIM Invoker transformation provides the input status for the INPUT_STATUS port of the Siebel EIM Read transformation based on whether the EIM task was invoked successfully.
- **SRT_EIM_ADDR_ORG.** A Siebel EIM Read transformation that extracts rows that were not successfully loaded into the Siebel base tables. The Siebel EIM Read transformation determines rows that were unsuccessfully loaded based on the IF_ROW_STAT column in the EIM table. The PowerCenter Integration Service then writes the output status to the OUTPUT_STATUS port of the Siebel EIM Read transformation and the read summary to the session log.
- **Account_FF.** The PowerCenter Integration Service writes extracted data to this flat file target.
- **CTI_CFG.** The PowerCenter Integration Service writes other extracted data to another flat file target.

Creating a Siebel EIM Read Transformation

You can create a Siebel EIM Read transformation in the Transformation Developer or the Mapping Designer.

1. Open the Transformation Developer.
Or, open the Mapping Designer.
2. Click Transformation > Create.
The Create Transformation dialog box appears.
3. Select Siebel EIM Read transformation from the list.
4. Enter a name for the transformation.
5. Click Create.
The wizard appears.
6. Enter the connection parameters.

The following table describes the connection parameters:

Connection Parameter	Description
Transport	Enter http or TCPIP. Default is TCPIP.
Encryption	Enter NONE or RSA. Default is NONE.
Compression	Enter NONE or zlib. Default is zlib.
Siebel Server Host	Siebel server host name.
SCBroker Port	Siebel Connection Broker port number.
Enterprise Server	Siebel Enterprise Server name.
Application Object Manager	Siebel application business object manager.
User Name	User name to connect to Siebel.
Password	Password to connect to Siebel.
Encoding	Encoding that the Designer uses to communicate with the Siebel server. By default, the Designer uses the UTF-8 encoding.

For more information about the connection parameters, contact your Siebel system administrator.

7. Click Connect to connect to Siebel.
8. Click Next.
9. To view the list of all EIM tables in Siebel, select EIM Tables in the Based on list.
Or, to view a list of Siebel base tables and corresponding EIM tables, select Base Tables in the Based On list.
10. Optionally, enter a filter condition to view particular Siebel base or EIM tables that meet the filter condition.
If the sort order of the database is case-sensitive, specify the EIM or base table name in uppercase letters.
11. Click Show Objects.
EIM tables appear if they are active in Siebel.
12. Select a base table to select all the corresponding EIM tables that you want to import.
13. Optionally, select an EIM table from the list of EIM tables.
The Table Fields area displays information about the selected EIM table.
14. Click Next.
15. Enter a name for the transformation.
If you use the Create Transformation dialog box to create the transformation, the transformation name specified in the Create Transformation dialog box overwrites the transformation name specified in the wizard.
16. Click Finish.
17. Click Done.

Understanding the Read Summary for the Siebel EIM Read Transformation

When the PowerCenter Integration Service extracts data from the Siebel EIM tables, it writes the read summary in the session log for each group of a partition. The read summary lists the following types of row counts for each group of a partition:

- Requested rows. Number of rows that the PowerCenter Integration Service receives from the input group to extract data.
- Rejected rows. Number of rows that the PowerCenter Integration Service does not extract because of an error.

CHAPTER 8

Siebel EIM Load Transformation

This chapter includes the following topics:

- [Siebel EIM Load Transformation Overview, 56](#)
- [Siebel EIM Load Transformation Components, 56](#)
- [Siebel EIM Load Transformation Groups and Ports, 57](#)
- [Using a Siebel EIM Load Transformation in a Mapping, 58](#)
- [Creating a Siebel EIM Load Transformation, 61](#)
- [Understanding the Load Summary for the Siebel EIM Load Transformation, 62](#)

Siebel EIM Load Transformation Overview

A Siebel EIM Load transformation is an active transformation that you can use to load bulk data from corporate data sources into the Siebel EIM tables.

When you run a session with a Siebel EIM Load transformation, the PowerCenter Integration Service connects to a Siebel database and loads data into the Siebel EIM tables. The PowerCenter Integration Service also writes the load summary in the session log for each group of a partition.

Siebel EIM Load Transformation Components

A Siebel EIM Load transformation contains the following tabs:

- **Transformation.** Enter the name and description of the transformation. The naming convention for a Siebel EIM Load transformation is *SLT_TransformationName*. You can also make the transformation reusable.
- **Ports.** View ports on the ports tab.
- **Properties.** View the properties on the Properties tab. Configure the tracing level property for the Siebel EIM Load transformation.

Note: You cannot modify the Ports tab in a Siebel EIM Load transformation.

Configuring Siebel EIM Load Transformation Properties

You can configure the tracing level of the session log on the Properties tab.

Siebel EIM Load Transformation Groups and Ports

A Siebel EIM Load transformation can contain multiple input groups. Each input group maps to the Siebel EIM table and contains ports that correspond to columns in the EIM table.

A Siebel EIM Load transformation contains the following output groups:

- STATUS
- CONTROL_FIELDS
- ERROR

The column name is stored in the Business Name attribute of the port. The description of the port contains more information about the port.

Note: You cannot edit the datatype, precision, or scale of any port.

STATUS Output Group

The STATUS output group contains a row for the load status of each partition, regardless of the number of input groups.

If the load for the entire partition was successful, the STATUS port displays 1. The PowerCenter Integration Service counts the number of non-fatal errors that occur while loading data into the EIM table. If the error count for a partition reaches or exceeds the error count limit, the PowerCenter Integration Service sets the status to 0 for that partition and does not process subsequent rows. If the error count for a partition reaches or exceeds the error threshold, the PowerCenter Integration Service fails the session.

CONTROL_FIELDS Output Group

The CONTROL_FIELDS group contains a row for each unique combination of EIM table name and batch number for the rows loaded into the EIM table. The CONTROL_FIELDS output group contains information for the rows that are successfully inserted or updated in Siebel EIM tables. If the EIM table contains multiple batch numbers, multiple output rows are created with the same Siebel EIM table name. The naming convention for CONTROL_FIELDS ports is <group name>_<field name>.

This group contains the following ports:

- CONTROL_FIELDS_EIM_TABLE. Contains the name of the EIM table into which the PowerCenter Integration Service loads data.
- CONTROL_FIELDS_BATCH_NUM. Contains the batch number for the rows that the PowerCenter Integration Service loads into the EIM table.

To improve session performance, PowerExchange for Siebel does not maintain or pass the port information if the CONTROL_FIELDS group is not linked in the mapping.

ERROR Output Group

The PowerCenter Integration Service passes the erroneous rows to the ERROR output group if any of the following non-fatal errors occur:

- Constraint violation errors while loading data into EIM tables
- Invalid batch number, Row ID, or LOV value
- Error while updating or deleting data from EIM tables

This group contains the following ports:

- **ERROR_EIM_TABLE.** Contains the name of the EIM table into which the PowerCenter Integration Service is loading data.
- **ERROR_ROW_ID.** Contains the row ID of the row for which the error occurred.
- **ERROR_BATCH_NUM.** Contains the batch number of the row for which the error occurred.
- **ERROR_MSG.** Contains detailed error message.

If a database error occurs, the error message appears in the following format:

```
SQLState|NativeErrorCode|ErrorMessage
```

If a non-database error occurs, the error message does not contain the SQLState and NativeErrorCode.

The error message appears in the following format:

```
||ErrorMessage
```

If multiple errors occur for a row while loading data into the EIM table, the PowerCenter Integration Service uses a new line character as a delimiter to concatenate the error messages in the ERROR_MSG port.

Using a Siebel EIM Load Transformation in a Mapping

You can use the Siebel EIM Load transformation in a mapping to load data from a corporate source into Siebel EIM tables and then invoke an EIM task to import rows from Siebel EIM tables into Siebel base tables or merge rows in Siebel base tables.

Importing Data from Siebel EIM Tables into Siebel Base Tables

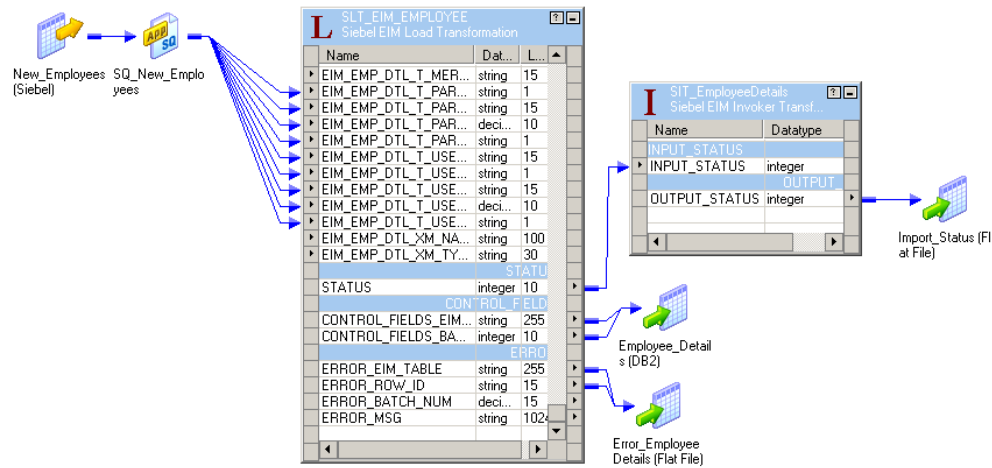
You can load data from a corporate source into Siebel EIM tables and then invoke an EIM task to import data from Siebel EIM tables into Siebel base tables.

For example, you need to import new employees data into Siebel base tables. After you configure the mapping to load the employee rows into an EIM table, you configure the Siebel EIM Invoker transformation to import the rows from EIM tables into the Siebel base table.

To configure the Siebel EIM Invoker transformation, configure the following values for parameters in the IFB file as in the following table:

Parameter	Value
TYPE	Import
BATCH	1

The following figure shows a mapping that loads data from a corporate source into Siebel base tables:



The mapping contains the following components:

- **New_Employees.** A source flat file that contains data to be loaded into EIM tables. The source ports provide values to the input group of the SLT_EIM_Employee transformation.
- **SQ_New_Employees.** An Application Source Qualifier that links the source fields to the ports in the input group of the SLT_EIM_Employee transformation.
- **SLT_EIM_Employee.** A Siebel EIM Load transformation that loads data from the source into the Siebel EIM tables. The PowerCenter Integration Service loads data into EIM tables for all ports of the input group linked in the SLT_EIM_Employee transformation. It also writes the batch number and EIM table name in the CONTROL_FIELDS output group for the rows loaded into EIM tables. The PowerCenter Integration Service writes the output status of the load operation to the STATUS output group and the rows that were not successfully loaded into EIM tables to the ERROR output group. It also writes the load summary to the session log.
- **Employee_Details.** A target flat file to which the PowerCenter Integration Service writes the data from the CONTROL_FIELDS group of the SLT_EIM_Employee transformation.
- **SIT_Employee_Details.** A Siebel EIM Invoker transformation that invokes the EIM task to write data from the Siebel EIM table to Siebel base tables. The Siebel EIM Invoker transformation invokes the specified EIM task if the input status equals '1.' After the PowerCenter Integration Service invokes the EIM task, it writes the output status to the OUTPUT_STATUS port of the SIT_Employee_Details transformation.
- **Error_Employee_Details.** A target flat file to which the PowerCenter Integration Service writes the erroneous rows from the ERROR output group of the SLT_EIM_Employee transformation.

Merging Data from Siebel EIM Tables into Siebel Base Tables

You can load data from a corporate source into a Siebel EIM table, and then invoke an EIM task that merges the data in the Siebel base tables. When you load data of rows marked for the merged operation, you must link the IF_ROW_MERGE_ID port in the source to the IF_ROW_MERGE_ID port of the Siebel EIM Load transformation. You must configure the IFB file for the merge operation and use this file to invoke the EIM task. The merge operation merges data from the Siebel EIM table into existing rows in a Siebel base table. The merged data from the Siebel EIM table replaces existing values in the Siebel base table.

For example, you create a mapping that extracts company data from an Account database, loads the company data into a Siebel EIM table, and then merges the company data into a Siebel base table.

The following table describes the parameters required to configure the IFB file for the EIM task:

Parameter	Value
TYPE	MERGE
BATCH	300

You configure the EIM task to merge rows in batch 300.

The following table displays the source information:

Row_ID	IF_ROW_BATCH_NUM	Company	IF_ROW_STAT	IF_ROW_MERGE_ID
1001	300	IBM	FOR_MERGE	NULL
1002	300	IBM Japan	FOR_MERGE	1001
1003	100	IBM Europe	FOR_MERGE	1001

You run the session.

The following table contains the EIM table information after the PowerCenter Integration Service loads the source data into the EIM table:

ROW_ID	IF_ROW_BATCH_NUM	Company	IF_ROW_STAT	IF_ROW_MERGE_ID
1001	300	IBM	FOR_MERGE	NULL
1002	300	IBM Japan	FOR_MERGE	1001
1003	100	IBM Europe	FOR_MERGE	1001

Next the PowerCenter Integration Service invokes the EIM task. Siebel EIM runs the EIM task based on the IFB file configuration and merges the rows into the Siebel base table. Siebel EIM merges the data of row 1002 into row 1001 and deletes row 1002 from the base table. It does not merge data of row 1003 into row 1001 because the batch number of row 1003 does not match the batch number configured in the IFB file.

The following table shows how Siebel EIM updates the status for these rows in the EIM table:

ROW_ID	IF_ROW_BATCH_NUM	Company	IF_ROW_STAT	IF_ROW_MERGE_ID
1001	300	IBM	MERGED_INTO	NULL
1002	300	IBM Japan	MERGED	1001
1003	100	IBM Europe	FOR_MERGE	1001

The status for row 1003 does not change because Siebel EIM does not merge data of row 1003 into row 1001.

Creating a Siebel EIM Load Transformation

You can create a Siebel EIM Load transformation in the Transformation Developer or the Mapping Designer.

1. Open the Transformation Developer.
Or, open the Mapping Designer.
2. Click Transformation > Create.
The Create Transformation dialog box appears.
3. Select Siebel EIM Load transformation from the list.
4. Enter a name for the Siebel EIM Load transformation.
5. Click Create.
The wizard appears.
6. Enter the connection parameters.

The following table describes the connection parameters:

Connection Parameter	Description
Transport	Enter http or TCPIP. Default is TCPIP.
Encryption	Enter NONE or RSA. Default is NONE.
Compression	Enter NONE or zlib. Default is zlib.
Siebel Server Host	Siebel server host name.
SCBroker Port	Siebel Connection Broker port number.
Enterprise Server	Siebel Enterprise Server name.
Application Object Manager	Siebel application business object manager.
User Name	User name to connect to Siebel.
Password	Password to connect to Siebel.
Encoding	Encoding that the Designer uses to communicate with the Siebel server. By default, the Designer uses the UTF-8 encoding.

For more information about the connection parameters, contact your Siebel system administrator.

7. Click Connect to connect to Siebel.
8. Click Next.
9. To view the list of all EIM tables in Siebel, select EIM Tables in the Based on list.
Or, to view a list of base tables and corresponding EIM tables in Siebel, select Base Tables in the Based on list.
10. Optionally, enter a filter condition to view particular Siebel base or EIM tables that meet the filter condition.
If the sort order of the database is case-sensitive, specify the EIM or base table name in uppercase letters.

11. Click Show Objects.
EIM tables appear if they are active in Siebel.
12. Select a base table to select all the corresponding EIM tables that you want to import.
13. Optionally, select an EIM table from the list of EIM tables.
The Table Fields area displays information about the selected EIM table.
If the LOV Bounded column is selected for a field, the Designer retrieves value for the LOV Type column.
14. Click Next.
15. Enter a name for the transformation.
If you use the Create Transformation dialog box to create the transformation, the transformation name specified in the Create Transformation dialog box overwrites the transformation name specified in the wizard.
16. Click Finish.
17. Click Done.

Rules and Guidelines for the Siebel EIM Load Transformation

Use the following rules and guidelines when you link a source to the Siebel EIM Load transformation:

- The batch number must be between 0 and 2147483647.
- If the EIM table contains the value in the LOV Type column and you configure the session to validate data against the LOV values, the source data must contain valid LOV values.

Understanding the Load Summary for the Siebel EIM Load Transformation

When the PowerCenter Integration Service loads data into the Siebel EIM tables, it writes the load summary in the session log for each group of a partition. The load summary lists the following types of row counts for each group of a partition:

- Requested rows. Number of rows that the PowerCenter Integration Service receives from the input group to load data into the Siebel EIM tables.
- Applied rows. Number of rows that the PowerCenter Integration Service tries to write to the Siebel EIM tables.
- Rejected rows. Number of rows that the PowerCenter Integration Service does not load into the Siebel EIM tables because of an error.
- Inserted rows. Number of rows that the PowerCenter Integration Service inserts into the Siebel EIM tables.
- Updated rows. Number of rows that the PowerCenter Integration Service updates in the Siebel EIM tables.
- Deleted rows. Number of rows that the PowerCenter Integration Service deletes from the Siebel EIM tables.
- Unprocessed rows. Number of rows that the PowerCenter Integration Service does not process because it reaches the error count limit for the partition.

CHAPTER 9

Siebel EIM Invoker Transformation

This chapter includes the following topics:

- [Siebel EIM Invoker Transformation Overview, 63](#)
- [Siebel EIM Invoker Transformation Components, 63](#)
- [Configuring an IFB File, 64](#)
- [Siebel EIM Invoker Transformation Groups and Ports, 64](#)
- [Using a Siebel EIM Invoker Transformation in a Mapping, 65](#)
- [Creating a Siebel EIM Invoker Transformation, 66](#)

Siebel EIM Invoker Transformation Overview

A Siebel EIM Invoker transformation is an active transformation that you can use to invoke an EIM task. The EIM task invokes one of the following business functions on a Siebel base table: export, import, merge, or delete. For example, you can invoke an EIM task that deletes rows from a Siebel base table.

The Siebel server requires a configuration file to run a business function. The configuration file specifies the Siebel EIM and base tables and the business function.

When you run a session with a Siebel EIM Invoker transformation, the PowerCenter Integration Service connects to the Siebel server and submits a request to invoke an EIM task. Siebel EIM invokes the specified EIM function.

Siebel EIM Invoker Transformation Components

A Siebel EIM Invoker transformation contains the following tabs:

- Transformation. Enter the name and description of the transformation. The naming convention for a Siebel EIM Invoker transformation is `SIT_TransformationName`. You can also make the transformation reusable.
- Ports. View ports on the Ports tab.

- Properties. View the properties on the Properties tab. Configure the tracing level property for the Siebel EIM Invoker transformation.
- Modify IFB File. Specify the IFB file path for the Siebel EIM Invoker transformation. Siebel EIM performs the import, export, merge, or delete operation based on the IFB file configuration. If a transformation is a reusable transformation, you cannot edit the transformation in the Mapping Designer.

Configuring an IFB File

You can configure the following parameters in an IFB file:

- EIM table. Specify the EIM table that provides the rows to be processed.
- Business function. Specify one of the following business functions: import, export, merge, or delete.
- Siebel base table. Specify the base table on which you want Siebel EIM to perform the business function.

For example, you can configure an IFB file to delete rows from Siebel base tables. The IFB file contains the following details:

```
[Siebel Interface Manager]
PROCESS = DELETE CONTACT
[DELETE CONTACT]
TYPE = DELETE
TABLE = EIM_CONTACT
CLEAR INTERFACE TABLE = TRUE
BATCH = 208
DELETE MATCHES = S_CONTACT, (FST_NAME='XYZ')
```

The IFB file is configured to delete the rows from the S_CONTACT base table that contains the value XYZ in the FST_NAME column and write the deleted rows to the EIM_CONTACT EIM table.

You can find the IFB in the following directory on the Siebel server machine:

```
<Siebel server installation>/ADMIN
```

Note: You can override the IFB file parameters by specifying the values in the EIM Job Parameters file.

Siebel EIM Invoker Transformation Groups and Ports

A Siebel EIM Invoker transformation contains the following input and output groups:

- INPUT_STATUS input group
- OUTPUT_STATUS output group

INPUT_STATUS Input Group

The PowerCenter Integration Service invokes an EIM task based on the value of the INPUT_STATUS port in the INPUT_STATUS group. The PowerCenter Integration Service does not invoke an EIM task if the value of this port is set to 0 or you do not connect a port that provides the input status to the INPUT_STATUS input port. If the value of the port is not set to any value set or to a value other than 0, the PowerCenter Integration Service invokes the EIM task.

When you use the Siebel EIM Invoker transformation in a mapping to invoke an EIM task, the INPUT_STATUS port receives the value from the first row of a source or from the output status of the previous transformation.

OUTPUT_STATUS Output Group

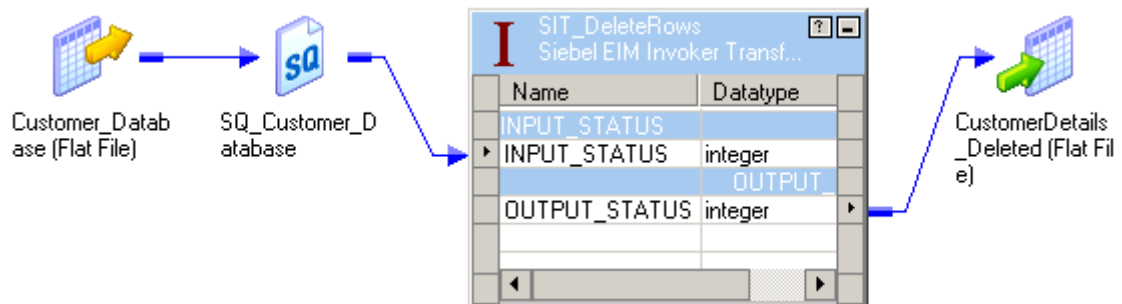
After invoking an EIM task, the PowerCenter Integration Service updates the OUTPUT_STATUS port of the OUTPUT_STATUS group. The PowerCenter Integration Service sets the value of the port to 1 if the EIM task is invoked successfully. The PowerCenter Integration Service sets the port to 0 if the EIM task is not invoked successfully.

Using a Siebel EIM Invoker Transformation in a Mapping

You can use the Siebel EIM Invoker transformation in a mapping to invoke an EIM task. When the PowerCenter Integration Service invokes the EIM task, Siebel EIM invokes the business function specified in the IFB file.

For example, you can configure the IFB file to delete rows from a Siebel base table.

The following figure shows a Siebel EIM Invoker transformation mapping that invokes an EIM task to delete rows from base tables:



The mapping contains the following components:

- Customer_Database. A source flat file that provides a value to the INPUT_STATUS port of the Siebel EIM Invoker transformation. When you run the session, the PowerCenter Integration Service invokes the EIM task if the value of the INPUT_STATUS port of the Siebel EIM Invoker transformation is not set to 0.
- SQ_Customer_Database. An Application Source Qualifier that links the input status in the source flat file to the INPUT_STATUS port of the Siebel EIM Invoker transformation.
- SIT_DeleteRows. A Siebel EIM Invoker transformation used in the mapping to invoke the EIM task that deletes the rows from the Siebel base table specified in the IFB file.
- CustomerDetails_Deleted. After the PowerCenter Integration Service invokes the EIM task, it writes the output status to the CustomerDetails_Deleted flat file target.

If the PowerCenter Integration Service invokes the EIM task successfully, Siebel EIM deletes the rows from the base table. When Siebel EIM attempts to delete a row from the Siebel base table, it updates the corresponding row in the EIM table to indicate the status of the deletion. When it successfully deletes a row,

Siebel EIM writes the batch number in the IF_ROW_BATCH_NUM column and the status 'DELETED' in the IF_ROW_STAT column of the EIM table.

Rules and Guidelines for the Siebel EIM Invoker Transformation

Use the following rules and guidelines when you create a mapping with a Siebel EIM Invoker transformation:

- Use a valid IFB file to invoke an EIM task.
- You must link the INPUT_STATUS port of the Siebel EIM Invoker transformation in a mapping. If you do not link the INPUT_STATUS port, the PowerCenter Integration Service does not invoke the EIM task.

Creating a Siebel EIM Invoker Transformation

You can create a Siebel EIM Invoker transformation in the Transformation Developer or the Mapping Designer.

1. Open the Transformation Developer.
Or, open a mapping in the Mapping Designer.

2. Click Transformation > Create.

3. Select Siebel EIM Invoker transformation.

4. Enter a name and click Create.

The wizard appears.

5. Enter the IFB file path.

6. Click Next.

7. Enter the name of the transformation.

If you use the Create Transformation dialog box to create a Siebel EIM Invoker transformation, the transformation name specified in the Create Transformation dialog box overwrites the transformation name specified in the wizard.

8. Click Finish.

9. Click Done.

CHAPTER 10

Siebel Sessions

This chapter includes the following topics:

- [Configuring Application Connection Objects for Siebel, 67](#)
- [Configuring Sessions for a Siebel Target, 69](#)
- [Configuring Sessions for a Siebel EIM Read Transformation, 70](#)
- [Configuring Sessions for a Siebel EIM Load Transformation, 71](#)
- [Configuring Sessions for a Siebel EIM Invoker Transformation, 74](#)
- [Error Handling, 76](#)
- [Troubleshooting Siebel Sessions, 77](#)

Configuring Application Connection Objects for Siebel

Configure an application connection to enable the PowerCenter Integration Service to connect to a Siebel object.

You can create application connections for the following types of objects:

- Siebel sources and targets
- Siebel EIM Invoker transformation
- Siebel EIM Load transformation
- Siebel EIM Read transformation

The PowerCenter Integration Service uses different application connection objects to connect to the Siebel repository and Siebel database.

Configuring Application Connections for Siebel Sources, Targets, and EIM Invoker Transformations

The Siebel sources, targets, and EIM Invoker transformations use the Siebel Connection application connection to connect to the Siebel repository. When you configure an application connection, you must specify the connection attributes for the Siebel repository.

1. In the Workflow Manager, click **Connections > Application**.
The Application Connection Browser dialog box appears.

2. Click New.
3. Select Siebel Connection from the list.
4. Click OK.

The Connection Object Definition dialog box appears.

5. Enter the connection parameters.

The following table describes the connection object parameters:

Connection Parameter	Description
Protocol	Protocol used to connect to Siebel. Specify the following protocol parameters: <ul style="list-style-type: none"> - Transport. Enter HTTP or TCP/IP. Default is TCP/IP. - Encryption. Enter NONE or RSA. Default is NONE. - Compression. Enter NONE or ZLIB. Default is ZLIB. Specify the parameters in the following format: <code>siebel[[.transport] [.encryption] [.compression]]]]</code>
User Name	User name to connect to Siebel.
Password	Password for the user name.
Siebel Server Host	Host name or IP address of the Siebel server. If you configure native load balancing, specify the virtual host name.
SCBroker Port	Siebel Connection Broker port number.
Enterprise Server	Enterprise Server name.
Application Object Manager	Siebel application business object manager.
Encoding	Encoding defined in the code page the PowerCenter Integration Service uses to communicate with the Siebel Server. Default is UTF-8.

For more information about connection parameters, contact your Siebel system administrator.

Configuring Application Connection for Siebel EIM Read and Load Transformations

The Siebel EIM Read and Load transformations use the Siebel Database Connection application connection to connect to the Siebel database. When you configure an application connection for the Siebel EIM Read or Load transformation, you must specify the connection attributes for the Siebel database.

To configure an application connection for the Siebel EIM Read or Load transformations:

1. In the Workflow Manager, click Connections > Application.
The Application Connection Browser dialog box appears.
2. Click New.
3. Select Siebel Database Connection from the list.
4. Click OK.

The Connection Object Definition dialog box appears.

5. Enter the connection parameters.

The following table describes the connection object parameters:

Connection Parameter	Description
User Name	User name to connect to Siebel.
Password	Password for the user name.
Connect String	ODBC data source name. Note: Use Informatica certified ODBC drivers for ODBC data source connection.
Connection Retry Period	Number of seconds the PowerCenter Integration Service attempts to reconnect to the database if the connection fails. If the PowerCenter Integration Service fails to connect to the database in the retry period, the session fails. If you set the connection retry period to 0, the PowerCenter Integration Service does not attempt to reconnect to the database if the connection fails. Default is 0.
Table Name Prefix	If required, configure the table name prefix to establish connection with the database. Default is Blank. Note: Enter the name of the Siebel database schema as the table name prefix when Oracle is the target database.

For more information about connection parameters, contact your Siebel system administrator.

Configuring Sessions for a Siebel Target

The following table describes the session properties for a Siebel target:

Session Property	Description
Duplicate Parent Row Handling	Select one of the following values to indicate how the PowerCenter Integration Service handles duplicate parent rows: <ul style="list-style-type: none">- First Row. The PowerCenter Integration Service writes the first row to the target and rejects subsequent rows with the same primary key.- Last Row. The PowerCenter Integration Service writes the last duplicate row to the target. The PowerCenter Integration Service rejects other rows with the same primary key.- Error. The PowerCenter Integration Service writes the first row to the target and increments the error count each time it encounters other rows with the same primary key. The session fails when the error count exceeds the error threshold.
Orphan Row Handling	Select one of the following values to indicate how the PowerCenter Integration Service handles orphan rows: <ul style="list-style-type: none">- Ignore. The PowerCenter Integration Service ignores orphan rows.- Error. The PowerCenter Integration Service treats orphan rows as error. The PowerCenter Integration Service increments the error count when it encounters an orphan row. The session fails when the error count exceeds the error threshold.

Session Property	Description
Cache Directory	Directory used to cache the Siebel target data. The default cache directory is \$PMCCacheDir.
Cache Size	Maximum buffer size the PowerCenter Integration Service uses to cache data before writing to the Siebel target. The default cache size to store the Siebel target data is 10 MB. The minimum value for cache size is 80 KB. You can set the cache size equal to 20 percent of the available memory of the system.

Note: You can view load statistics in the session log. The load summary in the Workflow Monitor does not display load statistics.

Configuring Sessions for a Siebel EIM Read Transformation

The following table describes the session properties that you can configure for a Siebel EIM Read transformation:

Session Property	Description
Rows to Read	<p>Select one of the following values to filter the rows to read from the Siebel EIM table:</p> <ul style="list-style-type: none"> - Successful. Reads all rows that have the IMPORTED, EXPORTED, MERGED, or DELETED status in the IF_ROW_STAT column. - Erroneous. Reads all rows that do not have the EXPORTED, IMPORTED, MERGED, DELETED, FOR_IMPORT, FOR_MERGE, or FOR_DELETE status in the IF_ROW_STAT column. - All. Reads all the rows. <p>The Successful and Erroneous options are applicable when the PowerCenter Integration Service extracts data after the load operation is executed.</p> <p>Default is blank. When you do not specify a value, the PowerCenter Integration Service uses the value set for the Rows to Read transformation property.</p>
Error Count Limit	Enter a limit for the error count. Set to '0' to disable the limit. Valid values include 0 to 2 ³¹ -1. Default is 0.

Session Property	Description
Filter	<p>Enter the filter condition to filter the rows read from a Siebel EIM table. You can configure a filter for multiple Siebel EIM tables. Use the following syntax:</p> <pre>[<Siebel EIM table name>:<SQL WHERE clause>]</pre> <p>For example, to apply the filters, enter the following filter values:</p> <pre>[EIM_AC_RSRC:ACR_NAME like 'Acr%'] [EIM_ACCNT_CUT:PARTY_UID='1-D1916' and AC_NAME like 'Acc%']</pre>
Delete Rows	<p>After the PowerCenter Integration Service extracts data from the Siebel EIM table, you can delete the rows from the Siebel EIM table.</p> <p>Select one of the following values to delete rows from the Siebel EIM table:</p> <ul style="list-style-type: none"> - Successful. Deletes all rows that have the IMPORTED, EXPORTED, MERGED, or DELETED status in the IF_ROW_STAT column. - All. Deletes all the rows. - None. Does not delete any row. <p>Default is None.</p>

Configuring Sessions for a Siebel EIM Load Transformation

The following table describes the session properties that you can configure for a Siebel EIM Load transformation:

Session Property	Description
Error Count Limit	Enter a limit for the error count. Set to '0' to disable the limit. Valid values include 0 to 2 ³¹ -1. Default is 0.
Truncate Table	Select to truncate the Siebel EIM table before the PowerCenter Integration Service loads data into the Siebel EIM table.
Enable LOV Validation	Select to validate the data against the list of values that correspond to the LOV type. By default, the PowerCenter Integration Service does not validate the data in the Siebel EIM table.
LOV Language	Specify the language code of the LOV data. Use the same format used by the Siebel EIM LOV parameter. Default is ENU.
EIM Table Load Type	<p>Select one of the following load types:</p> <ul style="list-style-type: none"> - Single Row. The PowerCenter Integration Service loads the data row-by-row into Siebel EIM tables. - Multiple Rows. The PowerCenter Integration Service loads the data for multiple rows in the same sequence as that of the source rows. The PowerCenter Integration Service processes consecutive rows with the Insert row type in batches, and processes rows with the Update or Delete row types one row at a time. <p>Default is Multiple Rows.</p>

Configuring the Load Type for the Siebel EIM Load Transformation

You can configure the load type for the Siebel EIM Load transformation. The load type determines if the PowerCenter Integration Service loads single rows or multiple rows.

When you configure the load type for the Siebel EIM Load transformation as single rows, the PowerCenter Integration Service loads the data row-by-row into Siebel EIM tables.

When you configure the load type for the Siebel EIM Load transformation as multiple rows, the PowerCenter Integration Service loads the data into Siebel EIM tables in batches based on the buffer block size. If you configure the session to treat source rows as data-driven and the load type as multiple rows, the PowerCenter Integration Service loads data into Siebel EIM tables based on the following conditions:

- If all source rows have the row type Insert, the PowerCenter Integration Service creates batches to load data into Siebel EIM tables. The PowerCenter Integration Service loads the rows into EIM tables in batches based on the buffer block size.
- If all source rows have the row type Update or Delete, the PowerCenter Integration Service loads the data row-by-row into Siebel EIM tables.
- If a source contains rows with multiple row types, the PowerCenter Integration Service loads the rows into EIM tables in the same sequence as that of the source rows. If the source contains rows with multiple row types, the PowerCenter Integration Service processes rows with the Update or Delete row type on a row-by-row basis. The PowerCenter Integration Service loads consecutive rows with the Insert row type in batches, based on the buffer block size.

For example, you configure the session to treat source rows as data driven and the load type as multiple rows.

The following table displays the source rows:

Row ID	Row Type
1	Update
2	Insert
3	Insert
4	Update
5	Delete
6	Insert
7	Update

The PowerCenter Integration Service processes rows in the following sequence:

1. Row with row ID 1.
2. Batch A that contains rows with the row type as Insert:

Batch ID	Row ID
A	2
A	3

3. Row with row ID 4.
4. Row with row ID 5.
5. Batch B that contains one row with the row type as Insert:

Batch ID	Row ID
B	6

6. Row with row ID 7.

Validating Data against Siebel List of Values

Siebel tables can contain List of Values (LOV) columns. Each LOV column corresponds to the LOV type that contains a list of valid values for a specific language. You can specify an LOV language for a Siebel EIM Load transformation. The LOV language of the Siebel EIM Load transformation must match the LOV language of the Siebel tables.

When you create Siebel EIM Load transformation and select a Siebel EIM table, you import the metadata for the columns of that EIM table. If the metadata contains the LOV type, you can configure the session to validate the data for that column against the list of valid values based on the LOV type and the LOV language.

For example, the EIM table EIM_ACCOUNT contains a column ACCNT_TYPE_CD. The LOV type of this column is CUTP_ACCOUNT_TYPE, which contains the following valid values:

- Billing
- Billing Aggregator
- Board
- Central Committee
- Customer
- Decentralized
- Insurance Company
- Intermediary
- Planner
- Service
- Service Aggregator

The LOV language for these values is English. When you load data into the EIM table, the PowerCenter Integration Service validates the data of the ACCNT_TYPE_CD column against these values. If the data of the column matches the values in the list, the PowerCenter Integration Service loads that data into EIM tables. Otherwise, it writes the erroneous rows into the ERROR output group of the Siebel EIM Load transformation.

Truncating EIM Tables

You can truncate the EIM table to delete the existing rows from the EIM table. If you run the same EIM task on an EIM table multiple times, you must truncate the EIM table before you process the same batch number again.

You can also truncate the existing rows from the EIM tables to enhance the performance of the EIM task. The PowerCenter Integration Service takes a longer time to perform actions on an EIM table that contains many rows. When possible, truncate the table to reduce the size of the table.

Configuring Sessions for a Siebel EIM Invoker Transformation

You can configure the following session properties for a Siebel EIM Invoker transformation:

Session Property	Description
Master Partition Index	You can configure a partition as a master partition by specifying the master partition index. Enter a value to greater than -1 to invoke one EIM task for all partitions. Enter the value -1 to invoke one EIM task for each partition.
Run Invoker when Input Status is	<p>You must specify a value for the Run Invoker When Input Status Is property if you set the Master Partition Index value greater than -1. Select one of the following options to invoke an EIM task based on the input status of the partitions:</p> <ul style="list-style-type: none"> - Success for All Partitions. The PowerCenter Integration Service invokes the EIM task when the values for the INPUT_STATUS port for all the partitions is 1. If the value for the INPUT_STATUS port of any of the partitions is not set to 1, the PowerCenter Integration Service does not invoke the EIM task. - Success for Any One Partition. The PowerCenter Integration Service invokes the EIM task if the value of the INPUT_STATUS port for any partition is 1. <p>Default is Success for All Partitions.</p>
Mode of Business Service	<p>Specify one of the following modes of the business service:</p> <ul style="list-style-type: none"> - Synchronous. The PowerCenter Integration Service invokes the EIM task immediately and waits for Siebel EIM to complete the EIM task before starting the next transformation in the mapping. After the EIM task completes, the PowerCenter Integration Service writes the task ID to the session log. - Asynchronous. The PowerCenter Integration Service sends a request to the Siebel server to invoke the EIM task and does not wait for the EIM task to complete before starting the next transformation in the mapping. The PowerCenter Integration Service also does not write the task ID to the session log. - Schedule. The PowerCenter Integration Service sends a request to the Siebel server to invoke the EIM task at the specified date and time. <p>Default is Asynchronous.</p>
Start Time	<p>Specify the start date and time for an EIM task in one of the following formats:</p> <ul style="list-style-type: none"> - MM/DD/YYYY HH:MM:SS - MM/DD/YY HH:MM:SS
EIM Job Parameters	<p>Specify the path and name of the file containing the list of EIM job parameters.</p> <p>This field is optional.</p>
Abort Session on Fatal Error	Select this option to abort the session when the PowerCenter Integration Service encounters a fatal error.
IFB File Path	<p>Specify the path and name of the IFB file in one of the following formats:</p> <ul style="list-style-type: none"> - Specify the absolute path and name of the IFB file. For example, specify: <code><Siebel server installation>/ADMIN/default.ifb</code> <p>Specify the relative directory path and name of the IFB file if you stored the file in the following directory or sub-directory:</p> <p><code><Siebel server installation directory>/ADMIN/</code></p> <p>For example, specify: <code>IFB_Folder/default.ifb</code></p>

EIM Tasks and Partitions

If the session contains multiple partitions, you can configure the PowerCenter Integration Service to run one EIM task for all partitions or one EIM task for each partition. You might choose to invoke one EIM task for all partitions if the PowerCenter Integration Service uses the same IFB file to load all partitions. You might choose to invoke one EIM task for each partition if the PowerCenter Integration Service uses a different IFB file to load each partition. If you choose to invoke one EIM task for all partitions, you can achieve better performance because the PowerCenter Integration Service sends only one request to the Siebel server to invoke the EIM task.

If you configure the session to invoke one EIM task for all partitions, the PowerCenter Integration Service invokes the EIM task only for the partition configured as master partition. The PowerCenter Integration Service sets the OUTPUT_STATUS of all other partitions to the same value as that of the OUTPUT_STATUS of the master partition.

When you configure the session to invoke an EIM task for each partition, the PowerCenter Integration Service invokes an EIM task for each partition based on the IFB file provided for that partition. For example, you assign batch numbers to data from multiple sources. You can partition the data based on the batch number, and then invoke different EIM tasks on each partition.

Note: The PowerCenter Integration Service fails the session when you configure the PowerCenter Integration Service to load multiple partitions and a session timeout occurs.

EIM Task and Business Service Modes

You can configure the following business service modes to invoke an EIM task on the Siebel server.

- Synchronous
- Asynchronous
- Schedule

The PowerCenter Integration Service sends a request to the Siebel server to invoke an EIM task. When the PowerCenter Integration Service sends the request to the Siebel server to invoke an EIM task, it writes the request ID to the session log.

Synchronous Business Service Mode

When you run a transformation in synchronous mode, the PowerCenter Integration Service invokes the EIM task and waits for the Siebel server to complete the EIM task before completing the remaining transformations in the mapping.

The Siebel server creates a log file when it runs the EIM task. The PowerCenter Integration Service writes the task ID of the EIM task and the log file name to the session log. It writes the log file name in the following format:

```
EIM_<EIM task ID>.log
```

You can use the task ID to track the EIM task status on the Siebel server and the log file name to locate the log file on the Siebel server.

Asynchronous Business Service Mode

When you run a transformation in asynchronous mode, the PowerCenter Integration Service sends a request to the Siebel server to invoke the EIM task. It does not wait for the Siebel server to complete the EIM task before starting the next transformation in the mapping. It also does not write the task ID of the EIM task to the session log.

Schedule Business Service Mode

When you run a transformation in schedule mode, the PowerCenter Integration Service invokes the EIM task on the configured date and time. It does not wait for the Siebel server to complete the EIM task. It does not write the task ID of the EIM task to the session log.

Creating an EIM Job Parameters File

Create an EIM job parameters file to configure the parameters for the EIM task. You can use this file to configure different parameters for one partition or multiple partitions. The value in the EIM job parameters file overrides any duplicate parameters in the IFB file.

For example, a session contains the following partitions and the corresponding batch numbers:

Partition Number	Batch Number
1	100
2	200

You can create one EIM job parameter file for each partition and add the parameters in the following format:

```
Batch=<batch number>
```

The EIM job parameters file for partition 1 contains the following parameter:

```
Batch=100
```

The EIM job parameters file for partition 2 contains the following parameter:

```
Batch=200
```

To configure an EIM job parameters file:

1. Create a .properties file.
2. Add parameters to the file that you need to invoke the EIM task.

Use the following format when you specify each parameter and value in the file:

```
<Parameter>=<Value>
```

3. Save the file to the machine running the PowerCenter Integration Service.

Error Handling

When you run a session, the PowerCenter Integration Service can encounter fatal or non-fatal errors.

Error Handling for Fatal Errors

If you extract data from or load data into Siebel EIM tables and a fatal error occurs, the PowerCenter Integration Service terminates the session.

The PowerCenter Integration Service terminates the session when you invoke an EIM task and any of the following fatal errors occur:

- Connection errors
- Invalid IFB file

- Failure to invoke an EIM task

If a session timeout occurs, the Siebel server continues to run the EIM task even though the PowerCenter Integration Service terminates the session. However, the PowerCenter Integration Service does not write the EIM task ID to the session log.

Error Handling for Non-Fatal Errors

If a non-fatal error occurs, the PowerCenter Integration Service increments the error count for each non-fatal error. The PowerCenter Integration Service continues to process the session until the PowerCenter Integration Service reaches the error count limit or error count threshold, or the session completes. If the error count reaches the error count limit, the PowerCenter Integration Service does not process subsequent rows and sets the output status of the Siebel EIM Load transformation to 0. If the error count reaches the error threshold, the PowerCenter Integration Service fails the session.

When the PowerCenter Integration Service reaches the error count limit or error count threshold, it does not roll back the previously loaded rows.

When the PowerCenter Integration Service loads rows into an EIM table in batches, it compares the error count against the error count limit and threshold after it processes each batch.

Troubleshooting Siebel Sessions

When I configure the PowerCenter Integration Service to load multiple partitions of a Siebel target, the session fails with the following error:

```
[OMRPC Request <Request ID> on connection <connection ID> was abandoned after
<milliseconds> ms because it timed out.]
```

To resolve the error:

1. Create the siebel.properties file and increase the values for the following connection parameters:
 - siebel.conmgr.sesstimeout. The Siebel server waits for the specified number of seconds before disconnecting an idle client session. The value for the parameter must be a positive integer. Default is 2700 seconds.
 - siebel.conmgr.poolsize. The Siebel server maintains a pool of open connections for each object manager process and shares the pool with all users of that process. Set the maximum number of connections that can be stored in a pool to a value between 1 and 499. Default is 2.
2. Copy the siebel.properties file to the following directory:


```
<PowerCenter Installation Directory>\server\bin\javalib
```
3. In the Administrator tool, append the directory path of the siebel.properties file in the Java SDK Classpath property for the PowerCenter Integration Service process.

When I configure the EIM task to run in Synchronous mode, the session fails with the following error:

```
[CMGR WARNING] Removing connection pool <connection string>/!1.cfc
```

To resolve the error:

1. Create the siebel.properties file.

2. Increase the value for the `siebel.conmgr.txtimeout` connection parameter.
The PowerCenter Integration Service waits for the configured number of milliseconds after sending a request to the Siebel Server. The value for the parameter must be a positive integer. Default is 600000 milliseconds.
3. Copy the `siebel.properties` file to the following directory:
`<PowerCenter Installation Directory>\server\bin\javalib`
4. In the Administrator tool, append the directory path of the `siebel.properties` file in the Java SDK ClassPath property for the PowerCenter Integration Service process.

APPENDIX A

Datatype Reference

This appendix includes the following topics:

- [Datatypes Overview, 79](#)
- [Siebel Business Component and Transformation Datatypes, 80](#)
- [Siebel Business Service Method and Transformation Datatypes, 81](#)
- [Siebel EIM and Transformation Datatypes, 81](#)

Datatypes Overview

PowerCenter uses the following datatypes in Siebel mappings:

- Siebel native datatypes. Siebel datatypes appear in Siebel definitions in a mapping. The PowerCenter Integration Service performs datatype conversions between transformation datatypes and the following Siebel datatypes:
 - Siebel business component field datatypes
 - Siebel business service method field datatypes
 - EIM table column datatypes
- Transformation datatypes. Set of datatypes that appear in the transformations. They are internal datatypes based on ANSI SQL-92 generic datatypes, which the PowerCenter Integration Service uses to move data across platforms. They appear in all transformations in a mapping.

When the PowerCenter Integration Service reads source data, it converts the native datatypes to the comparable transformation datatypes before transforming the data. When the PowerCenter Integration Service writes to a target, it converts the transformation datatypes to the comparable native datatypes.

Siebel Business Component and Transformation Datatypes

The following table lists the Siebel business component datatypes that PowerCenter supports and the corresponding transformation datatypes:

Siebel Business Component Datatype	Transformation Datatype	Description
DTYPE_BOOL	String	1 to 104,857,600 characters
DTYPE_CURRENCY	Decimal	Precision 1 to 28 digits, scale 0 to 28
DTYPE_DATE	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)
DTYPE_DATETIME	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)
DTYPE_ID	String	1 to 104,857,600 characters
DTYPE_INTEGER	Integer	Precision 1 to 10 digits, scale 0
DTYPE_NOTE	Text	1 to 104,857,600 characters
DTYPE_NUMBER	Decimal	Precision 1 to 28 digits, scale 0 to 28
DTYPE_PHONE	String	1 to 104,857,600 characters
DTYPE_TEXT ¹	Text	1 to 104,857,600 characters
DTYPE_TIME	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)
DTYPE_UTCDATETIME	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)

¹ PowerExchange for Siebel stores integration objects and hierarchy objects in DTYPE_TEXT in the PowerCenter repository.

Siebel Business Service Method and Transformation Datatypes

The following table lists the Siebel business service method datatypes that PowerCenter supports and the corresponding transformation datatypes:

Siebel Business Service Method Datatype	Transformation Datatype	Description
Date	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)
Hierarchy Object	String	1 to 104,857,600 characters
Integration Object	String	1 to 104,857,600 characters
Number	Integer	Precision 1 to 10 digits, scale 0
String	String	1 to 104,857,600 characters

Siebel EIM and Transformation Datatypes

The following table lists the Siebel EIM datatypes that PowerCenter supports and the corresponding transformation datatypes:

Siebel EIM Datatype	Description	Transformation Datatype	Description
Character	1 character	String	1 to 104,857,600 characters
Date	Jan 1, 1753 A.D. to Dec 31, 4712 A.D. (precision to microsecond)	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)
Date Time	Jan 1, 1753 00:00:00 A.D. to Dec 31, 4712 23:59:59 A.D. (precision to microsecond)	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)
Long	Up to 16 KB	Text	1 to 104,857,600 characters
Number	For non-integer columns, precision 1 to 22 digits, scale 0 to 7. For integer columns, precision 1 to 10 digits, scale 0.	Decimal	Precision 1 to 28 digits, scale 0 to 28
Time	00:00:00 to 23:59:59 (precision to microsecond)	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)

Siebel EIM Datatype	Description	Transformation Datatype	Description
UTC Date Time	Jan 1, 1753 00:00:00 A.D. to Dec 31, 4712 23:59:59 A.D. (precision to microsecond)	Date/Time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. (precision to nanosecond)
Varchar	Up to 2 KB	String	1 to 104,857,600 characters

APPENDIX B

Error Messages

This appendix includes the following topics:

- [Designer Messages, 83](#)
- [PowerCenter Integration Service Messages, 83](#)

Designer Messages

The Designer cannot find the Siebel .jar files. Copy the Siebel .jar files from Siebel client installation directory to the following directory: <PowerCenter Installation Directory>\clients\PowerCenterClient\client\bin\javalib.

Explanation: The Designer could not find the Siebel.jar and SiebelJL_enu.jar files in the following directory: <Siebel Installation Directory>\siebsrvr\CLASSES.

User Response: Copy the Siebel.jar and SiebelJL_enu.jar files to the following directory: <PowerCenter Installation Directory>\clients\PowerCenterClient\client\bin\javalib.

The Designer failed to connect to the Siebel server. Start the Siebel server, and then try to connect to the server.

Explanation: The Siebel server may be down or the specified connection parameters are incorrect.

User Response: Start the Siebel server, and then try to connect to the server.

PowerCenter Integration Service Messages

[1006 _SingleInvoke_InputStatusException]

The PowerCenter Integration Service did not invoke the EIM task because partition <partition number> is marked as unsuccessful.

Explanation: The value of the INPUT_STATUS port is not 1 for the partition.

User Response: Verify that the value of the INPUT_STATUS port is 1 for the partition or resolve the error in the partition that failed to process correctly.

[1016_BatchNumberZero]

The PowerCenter Integration Service cannot override the batch number for partition <partition number> in the IFB file when the batch number in the EIM job parameters file is set to 0.

Explanation: The batch number is 0 in the EIM job parameters file.

User Response: Specify a value other than 0 for the batch number in the EIM job parameters file.

[1032_SetBufferErr_Occured]

The PowerCenter Integration Service failed to write data in output port <output port name>.

Explanation: The PowerCenter Integration Service received incorrect data, or the data does not match the precision or datatype specified in the output port.

User Response: Enter the correct data, precision, and datatype for the output port.

[1033_InvalidMasterIndex]

The PowerCenter Integration Service set the Master Partition Index value to 0 because the specified value is greater than or equal to the partition count.

Explanation: The Master Partition Index value cannot be greater than or equal to the partition count.

User Response: Specify a Master Partition Index value that is less than the partition count.

[2001 _InvkFailed]

The PowerCenter Integration Service failed to invoke the EIM task for partition <partition number> because of the following Siebel error: <error>.

Explanation: The EIM task failed because of a connection error in Siebel, the PowerCenter Integration Service failed to invoke the Server Requests business service, or the Java Data Bean API did not find the Server Requests business service.

User Response: Verify that the siebel.conmgr.txttimeout connection parameter is valid in the siebel.properties file, the values specified for the Server Requests business service are correct, and the Server Requests business service exists in the Siebel repository.

[2006_AbortSession]

The PowerCenter Integration Service aborted the session due to a fatal error in partition <partition number>.

Explanation: The PowerCenter Integration Service encountered a fatal error while invoking the EIM task.

User Response: To resolve the error, verify the following have valid values: INPUT_STATUS port, connection parameters, parameters required to invoke the EIM task. Or, clear the Abort Session on Fatal Error session property.

[2021_InputStatus_SuccessAll_Exception]

The PowerCenter Integration Service did not invoke the EIM task because the Run Invoker When Input Status Is property is set to Success for All Partitions and at least one partition is marked as unsuccessful.

Explanation: The value of the INPUT_STATUS port is not 1 for all partitions.

User Response: Verify that the value of the INPUT_STATUS port is 1 for all partitions or resolve the error in the partition that failed to process correctly.

[4003_EIMPropertiesFileParsingException]

The PowerCenter Integration Service failed to parse the file <file name> specified in the EIM Job Parameters property for partition <partition number>. Error stack trace: <error stack trace>.

Explanation: The file does not have a valid format.

User Response: Verify the format of the file.

[34100]

The PowerCenter Integration Service failed to connect to the Siebel server.

Explanation: The PowerCenter Integration Service cannot connect to the Siebel server because of a connection error.

User Response: Verify that the Siebel server is not down, the PowerCenter Integration Service machine can access the Siebel server, and the connection parameters are correct.

[34102]

The PowerCenter Integration Service did not receive the locale message catalog <catalog name>.

Explanation: The PowerCenter Integration Service cannot find the locale message catalog in the following directory: <PowerCenter Installation Directory>\server\bin.

User Response: Verify that the local message catalog exists in the following directory: <PowerCenter Installation Directory>\server\bin. If it does not exist, reinstall the Server component of PowerExchange for Siebel.

[34105]

The PowerCenter Integration Service failed to execute the query to read rows from Siebel EIM table <table name>.

Explanation: The EIM table name, column name, table name prefix in the connection object, or format of the search filter condition may be incorrect.

User Response: Verify that the EIM table, EIM column, EIM table prefix names, and filter condition format are correct.

[34106]

The PowerCenter Integration Service failed to execute the query to delete rows from Siebel EIM table <table name>.

Explanation: The EIM table name, column names, table name prefix in the connection object may be incorrect.

User Response: Verify that the EIM table, EIM column, and EIM table prefix names are correct.

[34111]

The PowerCenter Integration Service encountered a memory allocation error.

Explanation: The PowerCenter Integration Service machine does not have enough memory.

User Response: Close some applications that are running on the PowerCenter Integration Service machine and try again.

[34119]

The CONTROL_FIELDS group did not receive a valid name for the Siebel EIM table <table name>.

Explanation: The CONTROL_FIELDS group either did not receive the name of the Siebel EIM table or the Siebel table name is in lowercase letters.

User Response: Specify the name of the Siebel EIM table in uppercase letters in the CONTROL_FIELDS group.

[41006]	The PowerCenter Integration Service encountered a memory allocation error when writing data to the EIM table.
Explanation:	The PowerCenter Integration Service machine does not have enough memory.
User Response:	Close some applications that are running on the PowerCenter Integration Service machine and try again.
[43030]	The PowerCenter Integration Service failed to execute the query to get the List of Values (LOV) type <type name> for table <table name>. Reason: <reason>.
Explanation:	The table name prefix in the connection property is invalid or the PowerCenter Integration Service lost the connection to the Siebel server.
User Response:	Verify that the table name prefix is correct and that the PowerCenter Integration Service can connect to the Siebel server.
[43032]	The PowerCenter Integration Service failed to create the database statement for the List of Values (LOV) type <type name>. Reason: <reason>.
Explanation:	The PowerCenter Integration Service is not connected to the target database.
User Response:	Verify that the PowerCenter Integration Service is connected to the target database. Or, refer to the reason for the error, and then review the documentation for the target database driver to resolve the error.
[44009]	The PowerCenter Integration Service failed to prepare the query for group <group name>. Reason: <reason>.
Explanation:	The PowerCenter Integration Service failed to prepare the query for the group.
User Response:	Specify the correct table name prefix in the connection properties. Or, refer to the reason for the error, and then review the documentation for the target database driver to resolve the error.
[44010]	The PowerCenter Integration Service failed to bind parameters to the database statement for group <group name>. Reason: <reason>.
Explanation:	The PowerCenter Integration Service cannot bind parameters to the statement for the group.
User Response:	Verify that the precision and scale of the ports in the transformation matches the corresponding precision and scale of the column in the target database. Or, refer to the reason for the error, and then review the documentation for the target database driver to resolve the error.
[44022]	The PowerCenter Integration Service cannot run the update, delete, or data-driven update strategy because the IF_ROW_BATCH_ID and ROW_ID fields are not linked in the mapping.
Explanation:	When the Treat Source Row As property is set to update, delete, or data-driven and the IF_ROW_BATCH_ID and ROW_ID fields are not linked in the mapping, the PowerCenter Integration Service cannot run the update strategy.
User Response:	Link the IF_ROW_BATCH_ID and ROW_ID fields in the mapping.
[44042]	The PowerCenter Integration Service failed to connect to the Siebel Database Server. Reason: <reason>.
Explanation:	The PowerCenter Integration Service lost the connection to the Siebel Database Server.

User Response:	Verify that the Siebel Database Server is not down, the PowerCenter Integration Service machine can access the Siebel Database Server, and the connection parameters are valid.
[44046]	The PowerCenter Integration Service failed to create the statement to truncate table <table name>. Reason: <reason>.
Explanation:	The PowerCenter Integration Service fails to create the statement to truncate the table <table name>.
User Response:	Verify that the connection to the Siebel Database Server is not down. Or, refer to the reason of the error, and then review the documentation of the target database driver to resolve the error.
[44047]	The PowerCenter Integration Service failed to execute the query to truncate table <table name>. Reason: <reason>.
Explanation:	The PowerCenter Integration Service failed to execute the query to truncate table <table name>.
User Response:	Verify that the PowerCenter Integration Service has permission to truncate a table. Or, refer to the reason for the error, and then review the documentation for the target database driver to resolve the error.
[44057]	The PowerCenter Integration Service could not set the Auto Commit mode for the session. Reason: <reason>.
Explanation:	The database user specified in the connection parameters does not have the right to set the Auto Commit mode for the session or another error occurred.
User Response:	Ask database administrator to grant the database user the right to set the Auto Commit mode for the session. Or, refer to the reason for the error, and then review the documentation for the target database driver to resolve the error.
[44065]	The PowerCenter Integration Service skips the entire row because it received the invalid value <invalid value> for the List of Values (LOV) type <type name>.
Explanation:	The value of the List of Values (LOV) type is invalid.
User Response:	Specify a valid value for the List of Values (LOV) type.
[SiebelRdr_1020]	The PowerCenter Integration Service failed to create the business component <business component name> used by business object <business object name>.
Explanation:	The business component is inactive. Or, the business component and the business object containing the business component are not linked in the repository.
User Response:	Contact Informatica Global Customer Support.

APPENDIX C

Glossary

application object manager

A Siebel component that manages all the Siebel objects in a Siebel application. The application object manager also hosts the Business Objects Layer and the Data Objects Layer.

business component

A logical entity that combines columns from base tables, extension tables, and joined tables into a single structure.

EIM job parameters file

A .properties file that contains parameters for the EIM task. The value in the EIM job parameters file overrides any duplicate parameters in the IFB file.

EIM task

The Siebel server task that inserts, extracts, deletes, and merges bulk data between Siebel EIM tables and Siebel base tables.

IFB file

The IFB file is an EIM configuration file of the Siebel system. The IFB file specifies the operations that the EIM task performs on the Siebel system.

integration object

A logical representation of external application data or Siebel data that corresponds to an external application.

List of Values

The List of Values (LOV) is a set of values stored in the Siebel database. Siebel tables can contain List of Values (LOV) columns and each LOV column corresponds to an LOV type. The LOV type contains a list of valid values for a specific language.

multi-value field

A field that can store more than one value at a time.

Siebel business service

A Siebel object used to implement reusable business logic that can be executed within the Object Manager.

Siebel Object Model

Represents Siebel object architecture, which includes a set of object definitions that are grouped into different layers depending on the functionality and characteristics of objects.

single-value field

A field that can store one value at a time.

INDEX

A

application connection
 configuring for PowerExchange for Siebel [67](#)
 configuring for Siebel EIM Read and Load transformations [68](#)
 configuring for Siebel sources, targets, and EIM Invoker transformations [67](#)

C

calculated fields
 Siebel sources [26](#)
CONTROL_FIELDS group
 rules and guidelines for Siebel EIM Read transformation [50](#)

D

datatypes
 PowerExchange for Siebel [79](#)

E

EIM job parameters file
 creating [76](#)
error messages
 Designer messages for PowerExchange for Siebel [83](#)
 PowerCenter Integration Service messages for PowerExchange for Siebel [83](#)

F

filters
 Siebel sources, Java Data Bean API [25](#)

I

IFB file
 configuring [64](#)
importing
 business component [26](#)
 calculated fields [26](#)
 custom fields [26](#)
 formatted fields [26](#)
 system fields [26](#)

J

Java components
 configuring for Siebel [19](#)

K

key relationship
 generating [36](#)
key values
 generating, Siebel sources [23](#)

M

multi-value fields
 Siebel sources [21](#)
 Siebel targets [34](#)

P

plug-ins
 registering [18](#)
 registering PowerExchange for Siebel [18](#)
 updating the registration [18](#)
PowerExchange for Siebel
 overview [10](#)
Properties tab
 Siebel EIM Load transformation [56](#)
 Siebel EIM Read transformation [48](#)

R

Read Transformation Properties tab
 Siebel EIM Read transformation [48](#)
registering
 plug-ins [18](#)

S

Siebel
 bulk data movement [13](#)
Siebel API libraries
 registering for PowerExchange for Siebel [19](#)
Siebel architecture
 Siebel Enterprise Applications [11](#)
Siebel business components
 importing [13](#)
Siebel Business Service transformation
 components [41](#)
 creating [45](#)
 groups [42](#)
 mapping example [42](#)
 working with [41](#)
Siebel EIM architecture
 illustration [13](#)
Siebel EIM Invoker transformation
 components [63](#)

Siebel EIM Invoker transformation (*continued*)

- creating [66](#)
- groups [64](#)
- invoking EIM task for partitions [75](#)
- invoking EIM task under business service modes [75](#)
- mapping example [65](#)
- overview [63](#)
- rules and guidelines [66](#)
- session processing [63](#)

Siebel EIM Load transformation

- components [56](#)
- creating [61](#)
- groups [57](#)
- List of Values [73](#)
- Load Type [72](#)
- mapping example [58](#)
- overview [56](#)
- properties [56](#)
- rules and guidelines [62](#)
- session processing [56](#)
- summary [62](#)
- truncating EIM tables [73](#)

Siebel EIM Read transformation

- components [47](#)
- creating [53](#)
- Filter [48](#)
- groups [49](#)
- mapping example [51](#)
- overview [47](#)
- properties [48](#)
- read transformation properties [48](#)
- rows to read [48](#)
- session processing [47](#)
- summary [55](#)

Siebel Enterprise Applications

- Siebel architecture [11](#)

Siebel Object Model

- description [11](#)
- illustration [11](#)

Siebel source definitions

- creating [38](#)
- editing [29](#)

Siebel source definitions (*continued*)

- importing [10](#)
- overview [21](#)
- reimporting [29](#)

Siebel target definitions

- editing [40](#)
- importing [38](#)
- overview [34](#)
- reimporting [40](#)
- troubleshooting [40](#)
- working with [34](#)

sort filter

- session-level [31](#)

source filter

- session-level [30](#)

T

transformation groups

- Siebel Business Service transformation [42](#)
- Siebel EIM Invoker transformation [64](#)
- Siebel EIM Load transformation [57](#)
- Siebel EIM Read transformation [49](#)

transformations

- Siebel Business Service transformation [41](#)
- Siebel EIM Invoker transformation [63](#)
- Siebel EIM Load Transformation [56](#)
- Siebel EIM Read transformation [47](#)

U

update strategy

- configuring [36](#)

V

view mode

- working with [32](#)