



Cisco Information Server Getting Started Guide

VERSION 6.2

Last Revision: December 6, 2013

Cisco Systems, Inc.
www.cisco.com

**Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.**

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2012 Cisco Systems, Inc. All rights reserved.



Contents

Preface 3

CHAPTER 1

Introduction 1-7

Overview 1-7

Cisco Information Server Architecture 1-8

CIS Resources 1-9

StudioModeling and Publishing 1-10

CHAPTER 2

Logging Into CIS Server 2-11

Connecting to CIS Server and Starting Studio 2-11

Studio Modeler and Resource Tree Overview 2-13

Viewing and Opening Resources 2-13

CHAPTER 3

Getting Started with CIS 3-15

About this Tutorial 3-15

Adding Data Sources 3-16

Creating a Folder 3-16

Adding the Orders Data Source 3-17

Viewing the Data Source Schema 3-20

Adding the Inventory Data Source 3-21

Adding the XML Data Source 3-22

Transforming XML Data in the XML Data Source 3-24

Selecting Output Columns in the XML Transformation 3-25

Building Simple Views 3-27

Building and Executing the Order Information View 3-28

Building and Executing the Supplier Information View 3-31

Building and Executing the Sales Information View 3-33

Creating a Composite View 3-35

Publishing Your Views 3-36

Publishing Views to CIS Databases 3-36

What You Learned in this Tutorial 3-37

CHAPTER 4

Sample Resources 4-39

- Sample Data Sources 4-39
- Sample Transformations 4-40
- Sample Definition Set 4-40
- Sample Published View 4-41
- Sample Views 4-41
- Sample SQL Script 4-41



Preface

Purpose

This guide is provided to familiarize you with basic functions of the Cisco Data Virtualization Platform.

The purpose of this guide includes:

- To demonstrate how you can use Studio to address your business needs
- To introduce the data modeling aspect of Studio

Audience

This book is intended for first-time Studio users who will work as data service oriented designers, developers, and architects interested in addressing business requirements, specification, and other business system issues.

Documentation Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands, keywords, GUI elements, and user-entered text appear in bold font.
<i>italic font</i>	Document titles, new or emphasized terms, and arguments for which you supply values are in italic font.
[]	Elements in square brackets are optional.
{ x y z }	Required alternative keywords are grouped in braces and separated by vertical bars.
[x y z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
courier font	Terminal sessions and information that the system displays appear in courier font.
<>	Nonprinting characters such as passwords are in angle brackets.
[]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Searching Across Multiple CIS PDF Documents

When you are looking for information in the CIS documentation set, you might want to search across multiple documents. You can use the free Adobe Reader to do this.



Note

If the options described below are not available in your version of Adobe Reader, please update it.

To search your CIS PDF documents

- Step 1** Open Adobe Reader.
- Step 2** From the File menu, choose Open. and open any PDF document.
If you cannot readily find a PDF document, browse to the folder shown in [Step 6](#) below.
- Step 3** From the Edit menu, choose Advanced Search.
- Step 4** Under Where would you like to search?, click All PDF Documents in.
- Step 5** Click My Documents and choose Browse for Location at the bottom of the drop-down list.
- Step 6** Browse to this location under Computer:
`C:\Program Files\Composite Software\CIS <version>\docs`
- Step 7** Enter the search term and click Search.
Acrobat lists all PDFs in the folder that contain the search string, and the number of occurrences in each.
- Step 8** Click the instance of the search term and its surrounding text to open the PDF to that page.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly *What's New in Cisco Product Documentation*, which also lists all new and revised Cisco technical documentation, at

<http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html>

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS Version 2.0.

Before contacting Support, be sure to have this information available:

- Type of request (help, bug report, or feature request)
- Product, current version, and patch level
- Case number, if one has been assigned

You can contact Technical Support at support@compositesw.com, (888) 210-9902, or <https://support.compositesw.com/>.

Document Change History

Table 1 provides the revision history for the *Getting Started Guide*.

Table 1 **Revision History**

Version Number	Issue Date	Status	Reason for Change
6.2	July 2012		No updates in this revision.
6.2 SP1	October 2012		No updates in this revision.
6.2 SP2	January 2013		No updates in this revision.
6.2 SP3	April 2013		Version update, including screen shots.
6.2 SP4	July 2013		Version update, including screen shots.
6.2 SP5	October 2013		Version update.
6.2 SP6	December 2013		No updates in this revision.





CHAPTER 1

Introduction

The Cisco Information Server (CIS) forms the core of the Data Virtualization Platform.

At build time, developers use the Cisco Information Server's easy-to-use development environment, Studio, with automated code generators, to create high-quality, semantically meaningful, standards-compliant views and data services. Rich tools enable complex federation and transformation functions. Standard adapters simplify access and publication development activities. The Manager controls features including security, metadata, and source code.

At run time, the Cisco Information Server's query engine securely queries, accesses, federates, abstracts and delivers data to consuming business solutions on demand. Multiple caching options provide additional speed and flexibility.

Topics covered in this chapter:

- [Overview, page 7](#)
- [Cisco Information Server Architecture, page 8](#)
- [CIS Resources, page 9](#)
- [StudioModeling and Publishing, page 10](#)

Overview

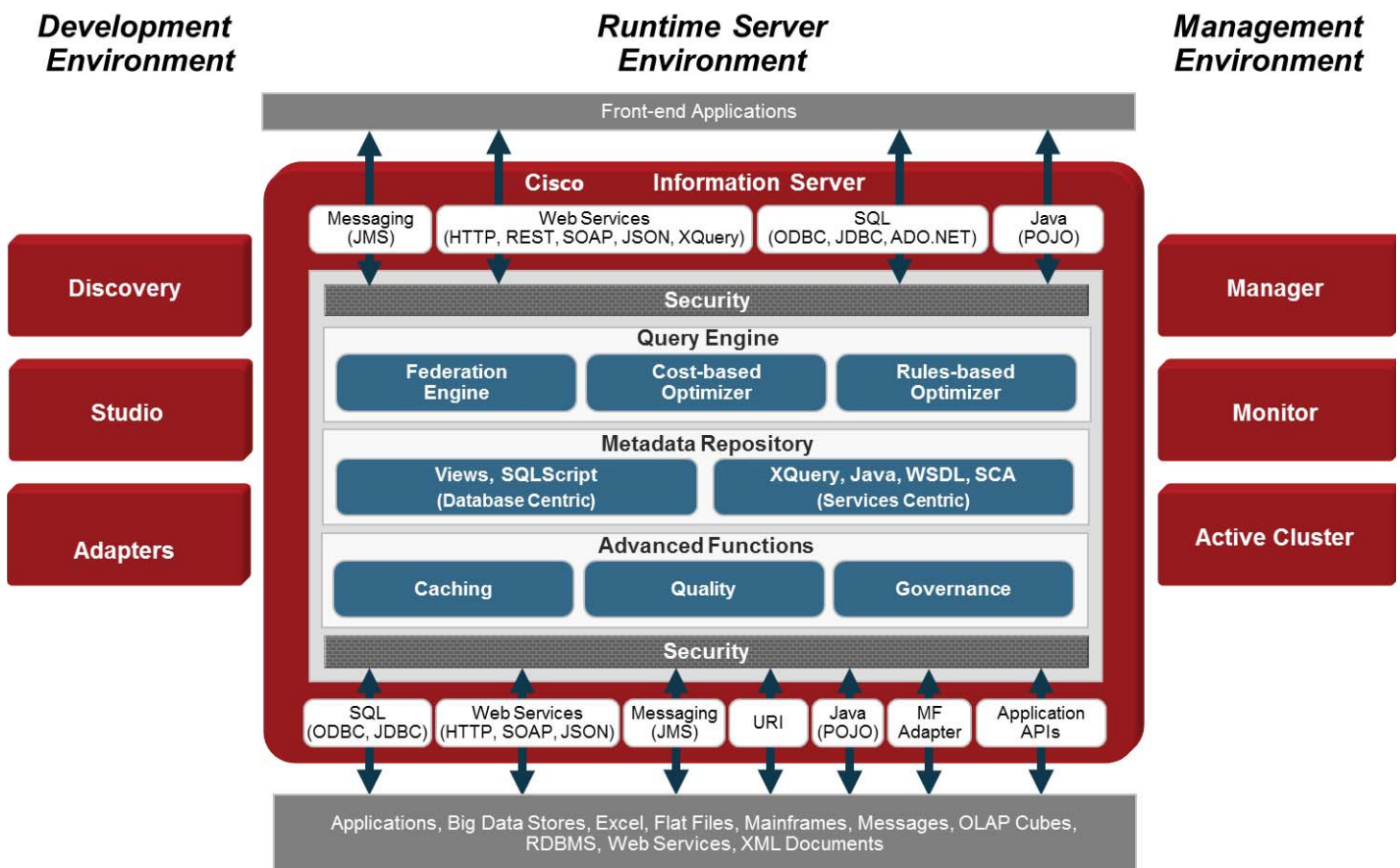
The Data Virtualization Platform is a suite of solutions that enables the definition of a virtual data layer to facilitate discovery, integration, and federation of disparate, distributed information sources. The CIS Server enables creation of a transparent, real-time interface to business information for business users and application developers. Designers using CIS create a securely managed unified view across files, databases, and packaged applications.

CIS supports a wide range of data sources including Oracle, Microsoft SQL Server, MySQL, DB2, Sybase, Informix, Netezza, Microsoft Access, Microsoft Excel, LDAP, flat files (including data in XML), and Web services.

The Cisco Information Server virtual data layer enables client applications to browse, query, update, and manage information gathered from across the enterprise and from any number or type of data sources.

Cisco Information Server Architecture

The CIS 6 is a data virtualization server that connects to existing data, federates disparate data, abstracts complex data, and delivers the information as data services. The server includes a graphical development environment that enables rapid design and development of database-centric objects including relational views and service-oriented objects. The Cisco Information Server also includes a complete set of management capabilities.



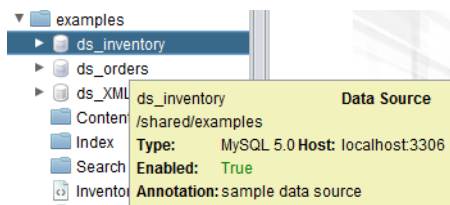
Product	Description	Books with more information
Cisco Information Server	CIS Server represents the core runtime environment that hosts various components such as the query engine and metadata repository.	<i>CIS User Guide.</i>
Studio	Studio provides an interface for data modeling, querying, transformation, and administration.	<i>CIS User Guide.</i>
Discovery	Discovery enables IT professionals to go beyond profiling to examine data, locate key entities and reveal hidden relationships in their enterprise data. You can use that knowledge to build rich data models for data virtualization and other information initiatives. The models allow you to access and show live data, making it easier to validate business requirements with end users.	<i>Discovery User Guide.</i>

Product	Description	Books with more information
Adaptors	<p>Adapters accelerate virtualized access to popular enterprise applications, relational and multi-dimensional data sources, “big data” stores, and Web content. The Adapters provide performance optimization capabilities and development acceleration. In addition to intelligently evaluating and leveraging underlying data source capabilities to ensure optimal federated query performance, Adapters provide:</p> <ul style="list-style-type: none"> • Certified Vendor-Specific Connectivity API – Access proprietary data using vendor-approved access methods. • Relational Representation of Source Data – Standardize data structures to accelerate development. • Business Canonical Abstraction of Source Data – Standardize data semantics to simplify development. • Vendor-specific Engineered Functions – Supercharge performance beyond vendor’s standard capabilities. 	Adapter Guides
Monitor	Monitor provides a comprehensive, real-time view of your Data Virtualization platform environment. Whether the environment is a single CIS or a cluster of instances, Monitor displays the indicators necessary to assess current conditions. If processes slow down or operations fail, your IT operations staff can use these insights to guide the actions required to maintain agreed service levels.	<i>Monitor Guide</i>
Active Cluster	Active Cluster allows you to scale your CIS deployments and maintain continuous availability of your data services. Active Cluster enables you to fulfill service level agreements by increasing capacity on demand, simplifying scaling, and improving manageability of your data services environment.	<i>Active Cluster Guide</i>

CIS Resources

CIS resources refers to the resources that are used for *data modeling* and building business solutions using CIS software. These resources are data sources, views, parameterized queries, SQL scripts, Java procedures, packaged queries, transformations, and CIS data services. Data stored in these resources are available in tabular or hierarchical format, and noted accordingly as either *tabular data* or *hierarchical data*.

Hover the pointer over the name of a resource and a tooltip displays the resource name, type, parent container, and other useful information like annotations, status, or target.



The parent container path combined with the resource name is the unique identifier for the invocation and reference to any CIS defined resource. For example you can have two tables, but because the parent container path is different and because the name and path used to refer to the resource are case-sensitive it is unique:

- /shared/examples/ds_inventory/inventorytransactions
- /shared/production/ds_inventory/InventoryTransactions

Resources include the following:

- data sources
- views
- procedures
- definition sets
- triggers
- CIS data services
- folders
- tabular and hierarchical data
- resources in context

For more information, see the *CIS User Guide*.

StudioModeling and Publishing

Metadata modeling and publishing in Cisco Information Server is a three-step process where you will:

- Introspect
- Model
- Publish

For details on these three processes, see the *CIS User Guide*.

Introspect

To introspect is to examine a physical data source and selecting only the specified resources from that data source for the sake of modeling in the CIS system. Introspection is a part of connecting to the data source. CIS lets you specify the physical data source you want to examine.

When you examine a data source, you do not have to choose the entire data source but can be selective about the data source resources you want to use for building a solution for your specific business.

Model

To model, you create, design, and edit views and procedures based on the introspected data sources.

You can experiment with different joins, columns, and constraints on views and procedures to find one that generates the result set that meets your business needs and drives your information integration project. You can also experiment with caching and join ordering to find the preferred performance profile, and subsequently store the view or procedure to be scheduled for batch reporting or to publish as a CIS database or Web service.

Publish

To publish you make the specified views and procedures available to client applications in the enterprise.



CHAPTER 2

Logging Into CIS Server

This chapter describes how to set up the work environment for using the software after installation.

- [Connecting to CIS Server and Starting Studio, page 11](#)
- [Studio Modeler and Resource Tree Overview, page 13](#)
- [Viewing and Opening Resources, page 13](#)

The installation process installs the server and other selected components in a specified location and starts its repository database (if the repository has been bundled with the installation).

CIS Server and Studio are available in a program group on the Start menu. CIS Server starts automatically after installation.

Connecting to CIS Server and Starting Studio

To successfully log into CIS Server, you should:

- Obtain a valid user name and password from your CIS administrator
- Know the name of the domain (composite or LDAP) to which you belong
- Know the name of the machine or the IP address where CIS Server is installed

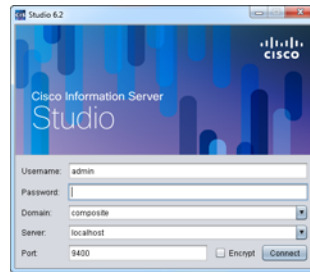
For details on installing the software, see the *CIS Installation Guide*.

This guide provides default values for the tutorial resources and sample data that can be installed with CIS.

Start Studio and log into CIS Server

Step 1 Select the Studio option.

Start > All Programs > Composite Software (version) > Studio > Start Studio (version).



Step 2 Type or select the following login information:

Field	Values used for the tutorial	Description
Username	admin (default value)	User name.
Password	admin	Password associated with your user name.
Domain	composite (default value)	Your domain from the drop-down list. This entry is used for authentication to connect to the server.
Server	localhost	The host machine for the server in the Server drop-down list. Localhost is the computer where Studio is running. Accept the default if Studio and the server are running on the same machine. If the server is not running on localhost, enter the IP address or the host name of the machine where the server is installed.
Port	9400 (default value)	Specify the HTTP base port number through which you can connect to the server. Change the default value manually, or it might have been changed because of the settings made for a previous installation.
Encrypt	blank	To connect to the server securely using SSL through HTTPS, check the Encrypt check box and all messages passed between the Studio session and CIS Server are encrypted. Otherwise HTTP clear text is used.

If Kerberos Single Sign-On has been enabled for your network then the SSO check box appears and you can check it to use your Windows login to connect with the Cisco Information Server. Otherwise proceed to login using user name and password authentication.

Step 3 Click **Connect**.

If the log-in credentials are validated successfully, you connect to CIS Server, and Studio opens for your modeling.

When the CIS Server is set to accept HTTPS secure mode communications only, a login attempted without the Encrypt check box results in a refused connection. The login error look similar to the following:

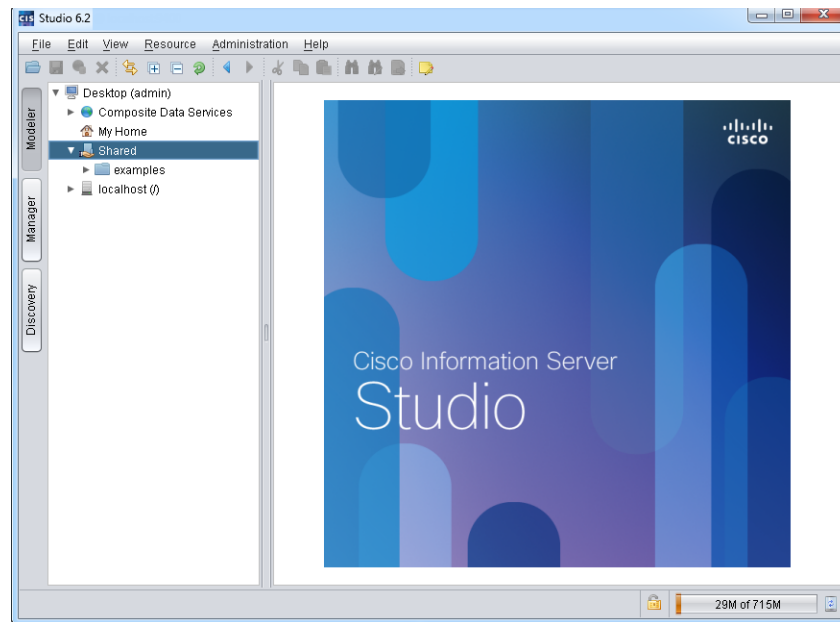
Failed to connect to http://localhost:9400/cdms/webapi; nested exception is:java.rmi.RemoteException: HTTP transport error. java.net.ConnectException...

Dismiss this error if received, and use the Encrypt check box to log in securely.

Studio Modeler and Resource Tree Overview

When you start Studio, the *Modeler* in Studio is displayed by default. The Modeler is the area where all data modeling activities take place. The left pane in the Modeler displays all available resources in a tree format, which is referred to as the *resource tree*.

Expand the node labeled **examples** that is inside the **Shared** folder



- **Desktop** — This area represents your (current user's) virtual work area in CIS Server, and is like the desktop for a personal computer's user interface.
- **Data Services** — This is the area visible to client applications that connect to CIS Server.
- **My Home** — This area represents your (current user's) workspace in CIS Server.
- **Shared** — This area is shared by all users in the system.
In My Home and Shared, you can create CIS resources.
- **examples** — This is a folder containing some sample resources to help you start using CIS.
- **<host machine>** — This area reflects the machine that hosts CIS Server.

For further details on the resource tree, see the *CIS User Guide*.

Viewing and Opening Resources

Some resources have been pre-created and are displayed in the resource tree. Only those users with administrative privileges can delete these resources. Others can view them and execute them.

To view the example resources

- Step 1** Expand the *Shared > examples* folder in the resource tree, and expand each node to see what is available.
- Step 2** Expand the *Data Services > Databases > examples* folder in the resource tree, and expand each node to see what is available.

To open a resource

- Step 1** Right-click the resource, and select **Open**.
- In the case of a leaf-node, you can also double-click it.



CHAPTER 3

Getting Started with CIS

This chapter provides a step-by-step tutorial that describes how to use the Modeler in Studio to create CIS data resources.

Topics covered in this chapter include:

- [About this Tutorial, page 15](#)
- [Adding Data Sources, page 16](#)
- [Transforming XML Data in the XML Data Source, page 24](#)
- [Selecting Output Columns in the XML Transformation, page 25](#)
- [Building Simple Views, page 27](#)
- [Creating a Composite View, page 35](#)
- [Publishing Your Views, page 36](#)
- [What You Learned in this Tutorial, page 37](#)

About this Tutorial

This tutorial teaches you how to build a unified solution to address a typical business scenario. For this example, you represent the three departments at ALPHA and build a unified view of your distributed business systems to address customer-reported issues. This example shows how to build the unified view using the following sequence of tasks:

Task		For instructions, see
1	Add data sources to CIS. You will add the three underlying data sources to the CIS metadata repository so you can query them. Each of these data sources will display as a data source in the resource tree.	Adding Data Sources, page 16
2	Build three simple views to retrieve data from the underlying data sources. Each of these views will extract specific information for you to address the business issue.	Building Simple Views, page 27
3	Join tables, and provide aliases for column names.	
4	Create the final unified view by combining the three individual views. This view unifies the separate pieces of information you retrieve through the three individual views you create.	Creating a Composite View, page 35

Task		For instructions, see
5	Create a CIS database. This database will be visible to client applications using JDBC and/or ODBC to connect to CIS Server.	To create a CIS data service of the type CIS database, page 36
6	Publish the unified views as CIS database tables. This unified view provides a single view solution for the business issue.	Publishing Your Views, page 36

For a quick tour of how to create and publish a REST service, see the tutorials in the *CIS User Guide*.

Adding Data Sources

Adding a data source means creating a CIS representation of the actual, underlying data source in the CIS metadata environment for data modeling. In some cases, you might add the entire data source and in others you might be selective about which resources in the data source that you want to add. The CIS representation of the data source consists of two things: the metadata that defines the data source and the information about connecting the data source to CIS Server.

Provided you have the WRITE privilege, you can add a data source to any area except Data Services in the resource tree.

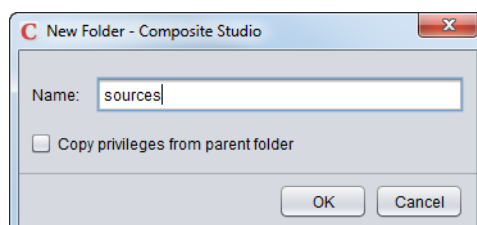
For details on privileges, see the *CIS Administration Guide*.

- [Creating a Folder, page 16](#)
- [Adding the Orders Data Source, page 17](#)
- [Viewing the Data Source Schema, page 20](#)
- [Adding the Inventory Data Source, page 21](#)
- [Adding the XML Data Source, page 22](#)

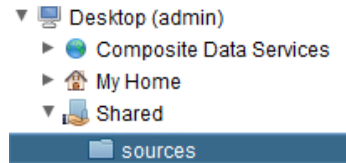
Creating a Folder

To create a folder named sources

- Step 1** Start Studio, if it is not currently running using the instructions in [Connecting to CIS Server and Starting Studio, page 11](#).
- Step 2** Right-click Shared, and select New Folder.
The New Folder window opens for you to name the new folder.
- Step 3** Type sources for the name in the input field, and click **OK**.



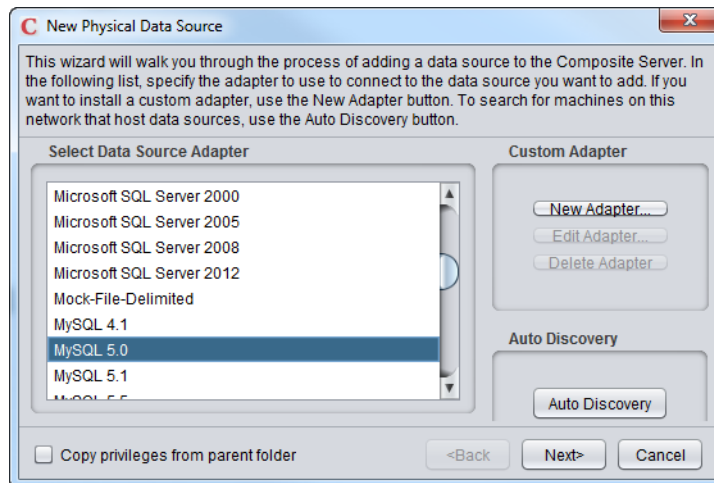
The newly created folder is added to the resource tree under **Shared**.



Adding the Orders Data Source

To add the orders data source

- Step 1** Right-click the `sources` folder and select New Data Source.
- Step 2** Scroll down and select MySQL 5.0 for the driver.



- Step 3** Click **Next**.
The window for specifying the data source information appears.
- Step 4** Select the Basic tab if it is not already selected.
- Step 5** Specify values for the following fields:

Field name	Tutorial Value	Description
Name	ds_orders	Type a unique name for the data source. When the process of adding this data source is complete, the name ds_orders will be displayed in the resource tree. ds_orders is the CIS representation of the underlying orders data source.
Host	localhost	Type the name or IP address of the machine where CIS Server is installed. If the server is installed on your local machine, you can type the machine name or the term localhost.
Port	9408	Type the port number. Use 9408 for the tutorial to access the MySQL server that is shipped with the CIS software. If you use an external MySQL server, the default port might be 3306.
Database Name	orders	CIS Server uses this entry to locate the underlying data source orders.
Login	tutorial	These entries are the user name and password to access the underlying data source.
Password	tutorial	

For details on the fields and the Advanced tab, see the *CIS User Guide*.

New Physical Data Source

Provide the connection information for this MySQL 5.0 data source.

Datasource Name

Name: ds_orders

Connection Information

Basic | Advanced

Host: localhost

Port: 9408

Database Name: orders

Login: tutorial

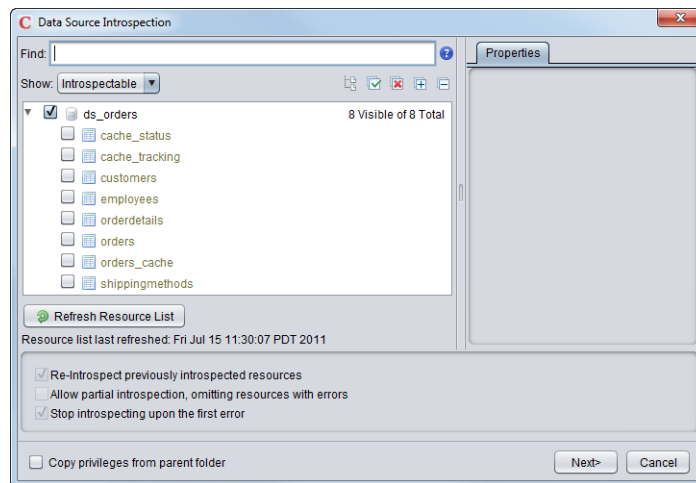
Password: *****

☒ Save Password

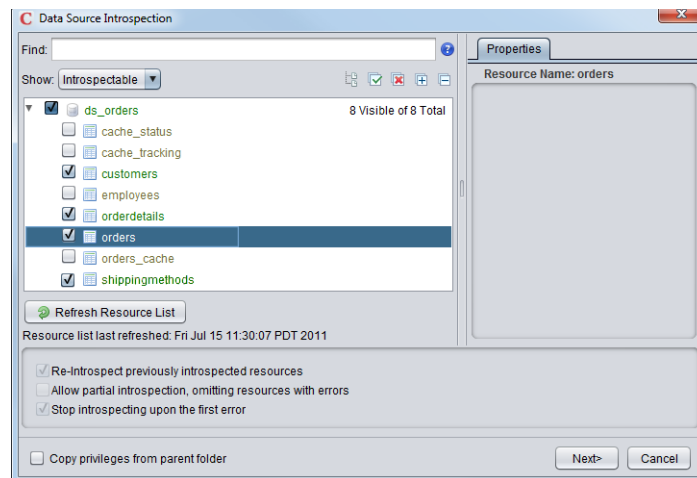
Pass-through Login: Disabled

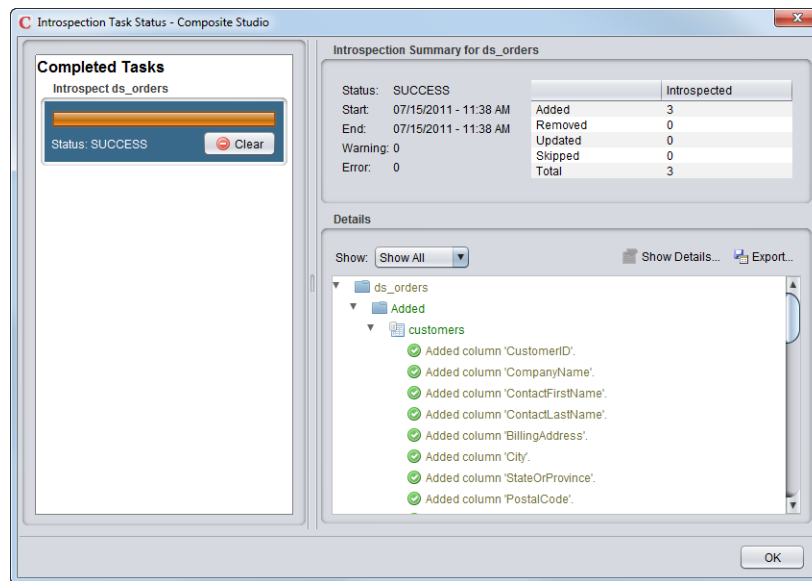
Transaction Isolation: Read Committed

<Back Create & Introspect Create & Close Cancel

Step 6 Click **Create & Introspect**.**Step 7** Select customers, orderdetails, orders, and shippingmethods.

These tables contain the information you need for customer contact and order details.

**Step 8** Click **Next**.**Step 9** Click **Finish**.



Step 10 Click **OK**.

A folder named `ds_orders` now appears under sources in the resource tree.

Viewing the Data Source Schema

To view the data source schema (optional)

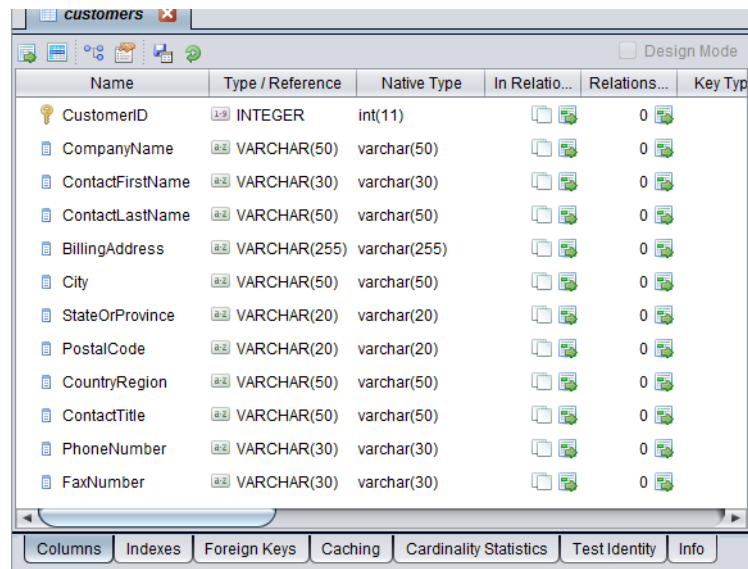
Step 1 Expand *Shared > sources > ds_orders*.

Each node within `ds_orders` represents a table. All of these tables are available for your modeling in Studio.

Step 2 Double-click the customers table name.

The table editor opens on the right. CIS data type and native data types are shown for each of the tables columns.

If you have Discovery, you will see several columns of information. If you run a Discovery process you can discover if a column has relationships or potential relationships to other data sources. From this information you can more effectively model your data environment.



Adding the Inventory Data Source

To add the inventory data source

- Step 1** Right-click the sources folder
- Step 2** Select New Data Source.
- Step 3** Scroll down, select MySQL 5.0 for the driver, and click **Next**.
- Step 4** Type values for the following fields:

Field name	Tutorial Value	Description
Name	ds_inventory	Type a unique name for the data source. When the process of adding this data source is complete, the name ds_inventory will be displayed in the resource tree. ds_inventory is the CIS representation of the underlying inventory data source.
Host	localhost	Type the name or IP address of the machine where CIS Server is installed. If the server is installed on your local machine, you can type the machine name or the term localhost.
Port	9408	Type the port number. Use 9408 for the tutorial to access the MySQL server that is shipped with the CIS software. If you use an external MySQL server, the default port might be 3306.
Database Name	inventory	CIS Server uses this entry to locate the underlying data source orders.
Login	tutorial	These entries are the user name and password to access the underlying data source.
Password	tutorial	

Step 5 Click **Create & Introspect**.

Step 6 Select the following tables:

- inventorytransactions
- products
- purchaseorders
- suppliers

These tables contain the information about product inventory transactions, purchase order details, and supplier contact information.

Step 7 Click **Next**.

Step 8 Click **Finish**.

Step 9 Click **OK**.

A folder named ds_inventory now appears under sources in the resource tree.

If you do not see the newly added data source in the resource tree, right-click the Desktop node, select Refresh, and look for it in sources.

Step 10 Expand *Shared > sources > ds_inventory*. Each node within ds_inventory represents a table you selected. Expand each node to see the columns in that table.

All of these tables and columns are available for your modeling in CIS.

Step 11 If you want to view the schema of this data source, follow the steps described for viewing the orders data source schema in [Viewing the Data Source Schema, page 20](#).

Adding the XML Data Source

The name of the XML data source used here is productCatalog.xml. It is located in the installation directory where the CIS software is installed. By default, on a Windows computer this location is:

```
<CIS_install_dir>\docs\examples
```

To add the XML data source

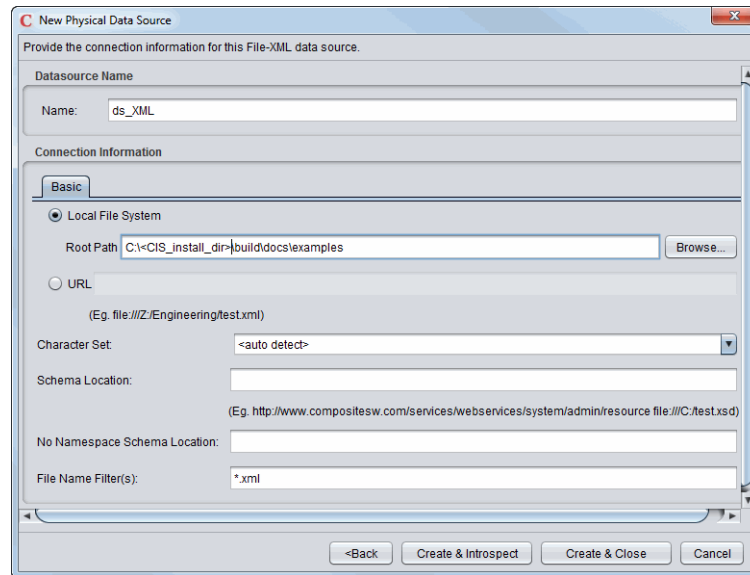
Step 1 Navigate to *Shared > sources* in the resource tree, right-click sources, and select New Data Source.

Step 2 Select File-XML as the adapter type, and click **Next**.

CIS displays the dialog for you to provide the connection information for this type of data source.

Step 3 In the New Physical Data Source window, type values for the following fields:

Field name	Tutorial Value	Description
Name	ds_XML	Type a unique name for the data source. This name is user-defined and will display in the resource tree when the process of adding the data source is complete.
Local File System	<CIS_install_dir>\docs\examples	Select the Local File System radio button and use the Browse button to locate the root path to the XML file, as in the following example: C:\Program Files\Composite Software\CIS6\docs\examples The root path does not include the name of the XML file, and only points to the directory of the file. You can also type the root path to the XML file in the Root Path field, instead of using the Browse button.
File Name Filter(s)	*.xml	



Step 4 Click **Create & Introspect**.

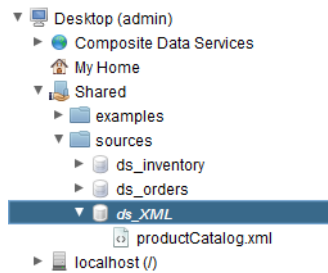
Step 5 Select productCatalog.xml.

Step 6 Click **Next**.

Step 7 Click **Finish**.

Step 8 Click **OK**.

The ds_XML data source is added to the resource tree at *Shared > sources*.



Transforming XML Data in the XML Data Source

This section gives quick steps for transforming our XML data source.

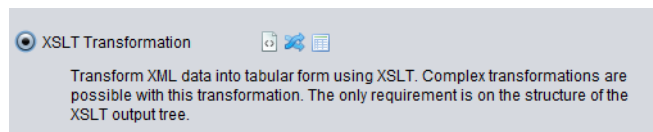
For more details on transformation, see the *CIS User Guide*.

The inventory and orders data sources contain tabular data (that is, relational tables), so they are ready to be queried in CIS's modeling environment. But the XML data source (productCatalog.xml) contains hierarchical data in XML format, so it must be transformed (or, flattened) into a table so you can use the data in your modeling.

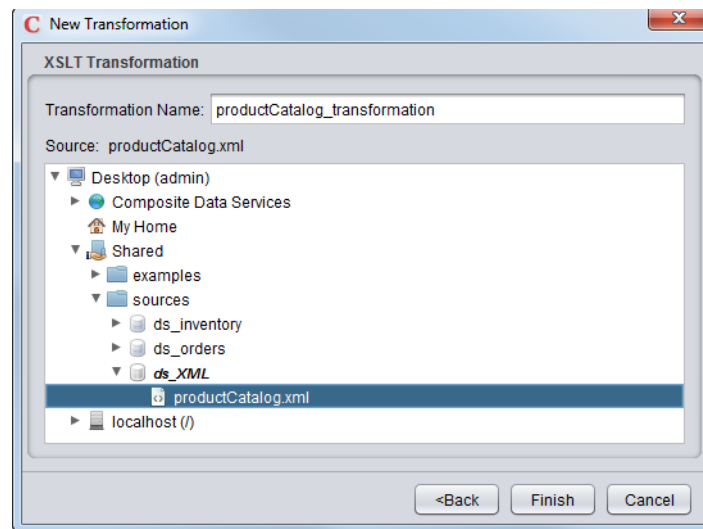
The process of converting hierarchical data into tabular form is called transformation in the system.

To transform the productCatalog.xml data source

-
- Step 1** Right click *Shared* > *sources*, and select New Transformation.
- Step 2** Select **XSLT Transformation**.

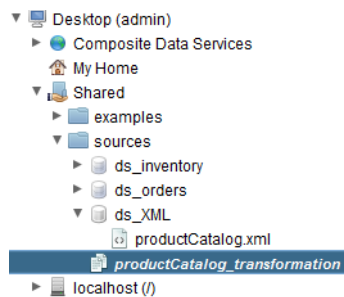


- Step 3** Click **Next**.
- Step 4** In the Transformation Name field, type productCatalog_transformation. This should always be a unique name.
- Step 5** Expand *Shared* > *sources* > *ds_XML*, and select productCatalog.xml.



Step 6 Click **Finish**.

The transformation is added to the resource tree under *Shared* > *sources*.



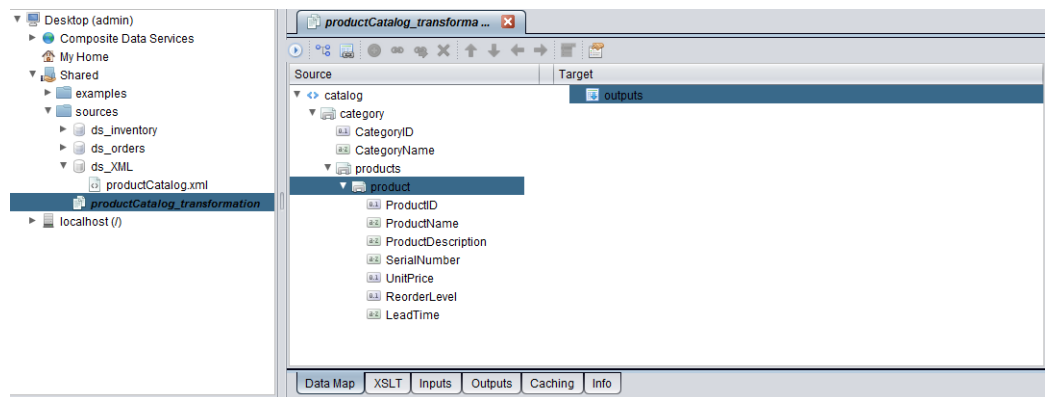
Selecting Output Columns in the XML Transformation

When the transformation of the XML data source is added to the resource tree, the editor opens on the right. The XML schema from the productCatalog.xml data source is displayed in its transformation (productCatalog_transformation) under the name catalog.

To select the columns that you need for your analysis

Step 1 In the workspace, expand all the nodes in catalog.

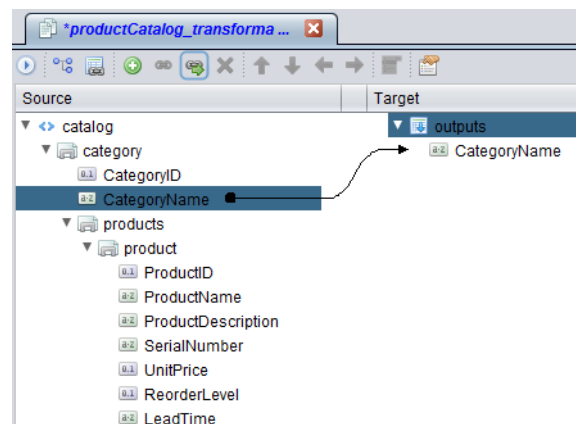
Selecting Output Columns in the XML Transformation



Step 2 Optionally, you can resize the column display for better viewing.

Step 3 Select **CategoryName** on the left.

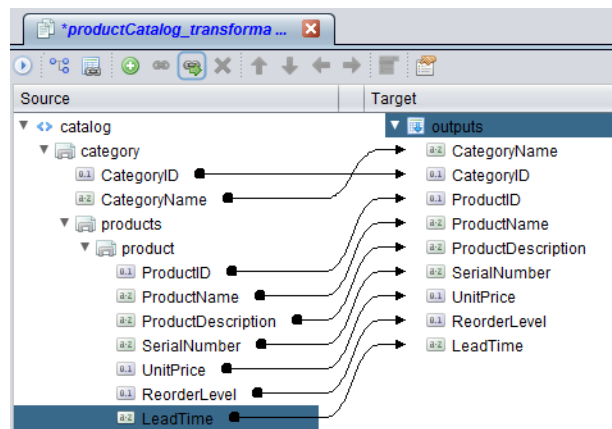
Step 4 Click **Create Link and Target**.




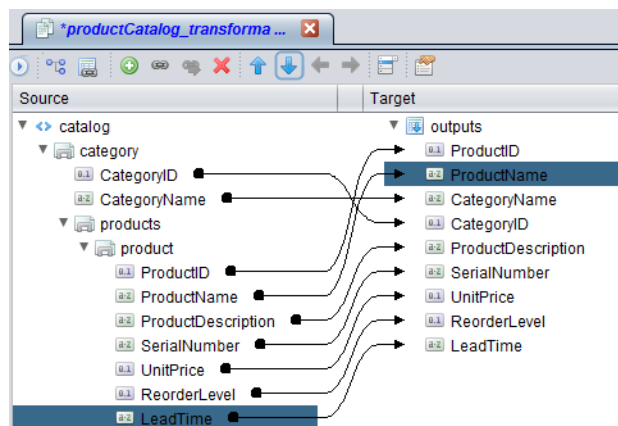
An entry is created with the name **CategoryName** under **outputs** in the **Target** column on the right, and the source and target are linked by a line.

Step 5 Select other columns, and click **Create Link and Target**.

For consecutive multi-selection, press the Shift key and make your selection. Otherwise, press the Ctrl key and make your selection. After selecting all the columns, the screen should look as follows:



Step 6 Move ProductID and ProductName up, by using the Move Up button , as follows:



Step 7 Save the transformation.

Step 8 Close the productCatalog_transformation tab.

You will query these sources to investigate the issue for ALPHA, to find out why its customer, Landmark Systems, did not receive ALPHA's product Widget 5 on the date promised (2/10/03). The next step is to build and execute views to obtain specific information about the activities in the Order, Purchase, and Sales departments at ALPHA.

Building Simple Views

This section describes how to create folders for organizing and storing your views, and how to create different views to explore the activities of the departments at ALPHA.

Building a view gives you a model, and executing it queries the relevant data sources and retrieves the specified data. For additional information on building and executing views, see the *CIS User Guide*.

In this section, you create the following three views to describe the data from the three departments at ALPHA:

- [Building and Executing the Order Information View, page 28](#)
- [Building and Executing the Supplier Information View, page 31](#)
- [Building and Executing the Sales Information View, page 33](#)

Building and Executing the Order Information View

To obtain information on the order and customer, you use the `ds_orders` data source which contains:

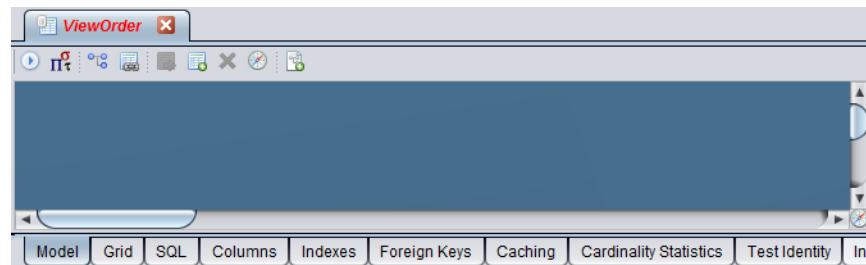
- Order information in the `orders` table.
- Details of each order in the `orderdetails` table.
- Customer information in the `customers` table.

You will include the tables `orders`, `orderdetails`, and `customers` in this view.

To build and execute the order information view

-
- Step 1** Right-click *Shared* > *sources*.
- Step 2** Select New View.
- Step 3** Type `viewOrder` as the name for the view.
- Step 4** Click **OK**.

When this view is added to the folder, the view editor opens for your use in the right pane of the Modeler.



The editor has the following tabs:

Tab	Description
Model	Use to assemble the tables.
Grid	Use to specify query constraints and the columns to include in the output when you execute your view.
SQL	Use to display the SQL for the view when you design the view in the Model and Grid panels. You can also type SQL in the SQL tab.
Columns	Use to list the columns to select for projection in the view execution result.
Indexes	Use to view reports on the index in the system tables, if the view is published.
Foreign Keys	Use to define foreign keys.

Tab	Description
Caching	Use to access the cache setting panel to configure the cache for storing the execution result of the current resource.
Cardinality Statistics	This panel provides an entry point to access the statistics setting panel where you can configure the statistics for the cached view. You have to configure the cache before configure the statistics. You can specify the following settings: statistics refresh mode, statistics refresh time-frame, and statistics clear mode.
Test Identity	This panel is provided for the row based security feature. In the Test Identity panel, you can specify the user and/or groups to be considered as the current user for purposes of simulating the results of row-level security. This facility does not alter any pass through credentials; data source connections remain under the identity of the current user.
Info	Use to provide annotations on the view.

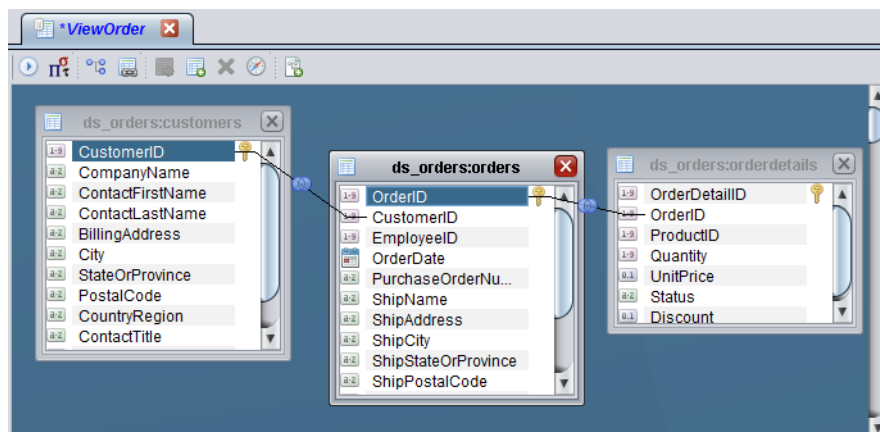
Step 5 Expand *Shared > sources* in the resource tree, and from the ds_orders data source drag the following tables into the Model panel in the editor:

- customers
- orderdetails
- orders

Step 6 Click OrderID in orders and drag it onto OrderID in orderdetails.

A line appears representing the inner join between the two tables.

Step 7 Join CustomerID in customers with CustomerID in orders.



For details on joins, see the *CIS User Guide*.

Step 8 Select the Grid panel.

The asterisk (*) in the first cell under Column indicates that all the columns in all the tables are selected for retrieval in the result set when the view is executed.

Step 9 To limit the columns in the result set:

- a. Click the first cell under Column.
- b. Select orderdetails.OrderID in the drop-down list.

c. Click more rows and select the following columns as you did in the preceding step:

- orderdetails.Status
- orderdetails.ProductID
- orderdetails.Discount
- orders.OrderDate
- customers.CompanyName
- customers.ContactFirstName
- customers.ContactLastName
- customers.PhoneNumber

Step 10 In the Alias column, click the cell next to `customers.ContactFirstName`.

Step 11 Type the alias `CustomerContactFirstName`.

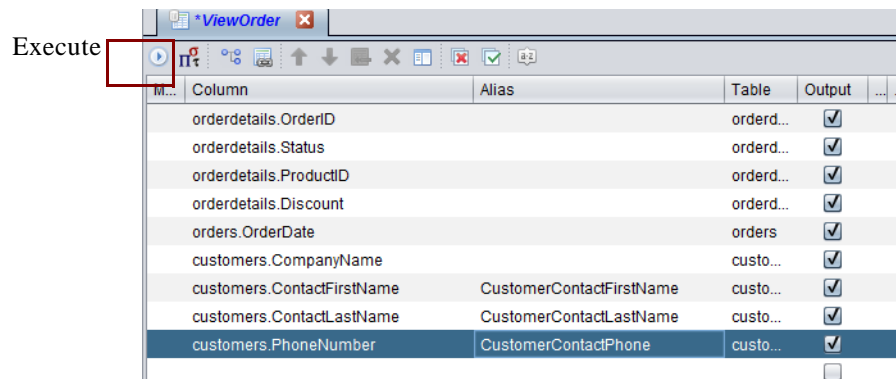
Step 12 Press the **Enter** key after typing the alias.

An alias makes a column name unique, which avoids conflicts with columns from other data sources having the same name.

Step 13 Type the alias for each of the following columns.

Column	Alias to type
customers.ContactLastName	CustomerContactLastName
customers.PhoneNumber	CustomerContactPhone

The asterisks next to the View Order indicates that you have made changes but have not saved them.



Step 14 Save the view.

Step 15 Optionally, you can view the SQL for this view by selecting the SQL tab. After you view the SQL, return to the modeling area by selecting the Model tab.

Typing or editing the SQL in the SQL panel invalidates the design made in the Model and Grid panels.

Step 16 Execute the view by clicking the **Execute** button.

The Result panel opens and displays the result of the view's SQL execution.

Step 17 In the result displayed in the Result panel, identify the row for `OrderID = 24`, which has the following data:

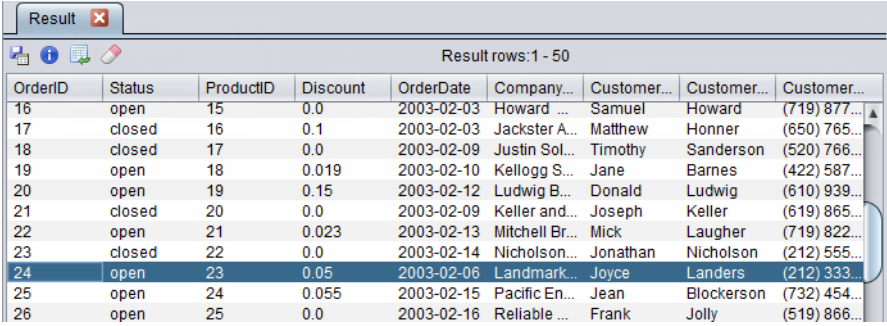

```

OrderID: 24
Status: open
ProductID: 23
Discount: 0.05
OrderDate: 2003-02-06
CompanyName: Landmark Systems
CustomerContactFirstName: Joyce
CustomerContactLastName: Landers
CustomerPhone: (212) 333-1000

```

This result set contains the basic information for the Order department about the order identification, order date, product identification, and customer that are all relevant for the current example.

The following screen shows the view execution results displayed in the Result panel.



OrderID	Status	ProductID	Discount	OrderDate	Company...	Customer...	Customer...	Customer...
16	open	15	0.0	2003-02-03	Howard ...	Samuel	Howard	(719) 877...
17	closed	16	0.1	2003-02-03	Jackster A...	Matthew	Honner	(650) 765...
18	closed	17	0.0	2003-02-09	Justin Sol...	Timothy	Sanderson	(520) 766...
19	open	18	0.019	2003-02-10	Kellogg S...	Jane	Barnes	(422) 587...
20	open	19	0.15	2003-02-12	Ludwig B...	Donald	Ludwig	(610) 939...
21	closed	20	0.0	2003-02-09	Keller and...	Joseph	Keller	(619) 865...
22	open	21	0.023	2003-02-13	Mitchell Br...	Mick	Laughner	(719) 822...
23	closed	22	0.0	2003-02-14	Nicholson...	Jonathan	Nicholson	(212) 555...
24	open	23	0.05	2003-02-06	Landmark...	Joyce	Landers	(212) 333...
25	open	24	0.055	2003-02-15	Pacific En...	Jean	Blockerson	(732) 454...
26	open	25	0.0	2003-02-16	Reliable ...	Frank	Jolly	(519) 866...

Step 18 Close the ViewOrder view tab.

Building and Executing the Supplier Information View

To obtain information on the purchase order and supplier, you use this information in the ds_inventory data source:

- Product transaction information in the inventorytransactions table.
- Purchase details in the purchaseorders table.
- Supplier information in the suppliers table.

To build and execute the supplier information view

-
- Step 1** Right-click *Shared > sources* in the resource tree.
- Step 2** Create a new view named `viewSupplier`.
- Step 3** Drag the following tables from ds_inventory and drop them into the Model panel of ViewSupplier:
- inventorytransactions
 - purchaseorders
 - suppliers
- Step 4** Join the following columns to each other:

Table Name	Column Name	Join	Table Name	Column Name
suppliers	SupplierID	to	purchaseorders	SupplierID
purchaseorders	PurchaseOrderID	to	inventorytransactions	PurchaseOrderID

Step 5 In the Grid panel, limit the query to the following columns as described in [Building and Executing the Order Information View, page 28](#):

```
inventorytransactions.ProductID
inventorytransactions.TransactionID
purchaseorders.DateRequired
purchaseorders.DatePromised
purchaseorders.ShipDate
purchaseorders.SupplierID
suppliers.SupplierName
suppliers.ContactName
suppliers.PhoneNumber
```

Step 6 Provide aliases for the following column names:

Column Name	Alias to type
suppliers.ContactName	SupplierContactName
suppliers.PhoneNumber	SupplierPhoneNumber

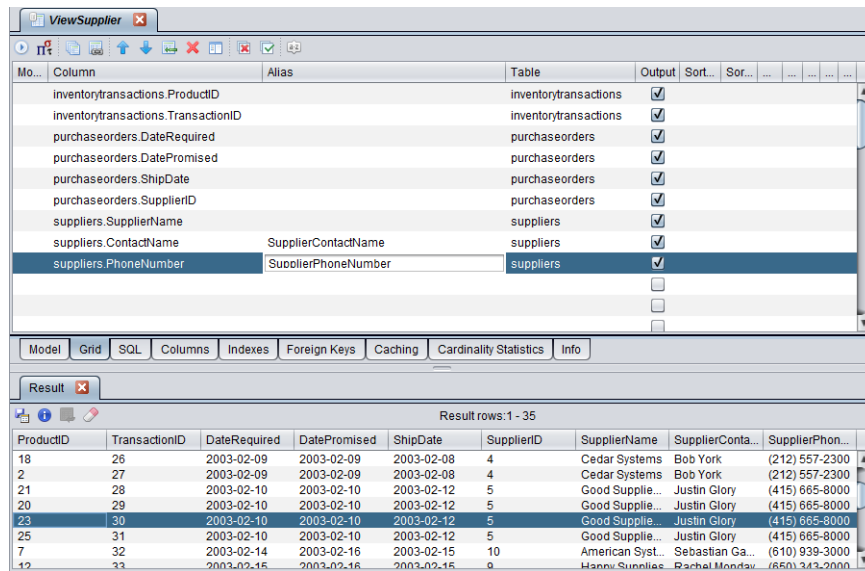
Step 7 Save the view.

Step 8 Execute the view to see the result.

Step 9 In the result set displayed in the Result panel, identify the row for `ProductID = 23` and `Transaction ID = 30` and which has the following data:

```
ProductID: 23
TransactionID: 30
DateRequired: 2003-02-10
DatePromised: 2003-02-10
ShipDate: 2003-02-12
SupplierName: Good Supplies International
SupplierID: 5
SupplierContactName: Justin Glory
SupplierPhoneNumber: 415-665-8000
```

This result set contains all the information the Purchase department needs about the purchase order and supplier that are relevant for the current example. The supplier did not deliver the order on the promised date, which is 2003-02-10. Additionally, the supplier shipped the order late (on 2003-02-12).



Step 10 Close the ViewSupplier view tab.

Building and Executing the Sales Information View

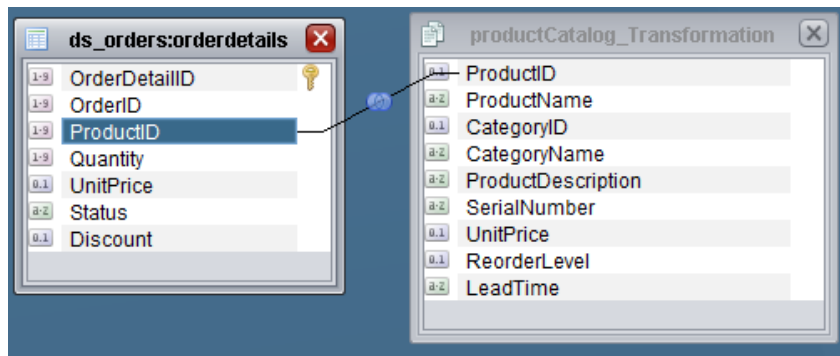
To obtain product sales information, you use the orders data source and the transformation productCatalog_transformation.

- The orders data source contains order details in the orderdetails table.
- The transformation productCatalog_transformation contains product catalog information.

To build and execute the sales information view

- Step 1** Right-click *Shared > sources* in the resource tree.
- Step 2** Create a new view named ViewSales.
- Step 3** From *Shared > sources*, expand the *ds_orders* node and drag the *orderdetails* table into the Model panel.
- Step 4** From *Desktop > Shared > examples*, drag the *productCatalog_transformation* into the Model panel.
- Step 5** Join *ProductID* in *orderdetails* with *ProductID* in *productCatalog_transformation*.

The join between two different types of resources, one derived from a relational table, and the other from an XML data source.



Step 6 In the Grid panel, limit the query to the following columns:

```
productCatalog_transformation.*
orderdetails.Status
orderdetails.Discount
```

Step 7 Save the view.

Step 8 Execute the view.

Step 9 In the result set displayed in the Result panel, locate one of the three rows where `ProductID = 23` with the following data:

```
ProductID: 23
ProductName: Widget 5
CategoryID: 7
CategoryName: Networking
ProductDescription: WidgetModel
SerialNumber: 5
ReorderLevel: 5
LeadTime: 1 Day
Status: open
Discount: 0.05
```

These results contain information about the product name, the lead time for product delivery, and other data relevant for the current example.

Step 10 Close the ViewSales tab.

After adding the data sources to the CIS metadata environment, you created three views as follows:


- View 1 to obtain order information for the Order department.
By executing this view, you were able to view the activities of the order department at ALPHA. You learned about the product status, order date, and customer contact information.
- View 2 to obtain supplier information for the Purchase department.
By executing this view, you were able to learn about ALPHA's purchase department's interactions with the customer and the supplier.
- View 3 to obtain sales information for the Sales department.
By executing this view, you were able to use the sales department's record and noted the lead time for product delivery.

You will query these views and create a single unified view representing the solution for the customer-reported issue at ALPHA.

Creating a Composite View

For the Sales department at ALPHA, you can build a single, composite view of ALPHA's business data from the three individual views you have created: ViewOrder, ViewSupplier, and ViewSales. There is no need to access the data sources any more.

To create the composite view

-
- Step 1** Expand *Shared > sources* in the resource tree.
- Step 2** Create a new view named `CompositeView`.
- Step 3** Drag the following views and drop them into the Model panel of `CompositeView`:
- ViewOrder
 - ViewSales
 - ViewSupplier
- Step 4** Join `ProductID` in ViewOrder with `ProductID` in ViewSales.
- Step 5** Join `ProductID` in ViewSales with `ProductID` in ViewSupplier.
- Step 6** In the Grid panel, limit the query to the following columns:
- ```
ViewOrder.*
ViewSales.ProductName
ViewSales.LeadTime
ViewSupplier.TransactionID
ViewSupplier.DateRequired
ViewSupplier.DatePromised
ViewSupplier.ShipDate
ViewSupplier.SupplierID
ViewSupplier.SupplierName
ViewSupplier.SupplierContactName
ViewSupplier.SupplierPhoneNumber
```
- Step 7** Save the view.
- Step 8** Execute the view. Result rows 1-50 are displayed.
- Step 9** In the result set displayed in the **Result** panel, look for a row where `OrderID = 24` and `Transaction ID = 30`. There are multiple rows that fit this criteria in the result rows 1-150.
- Step 10** Click Load More Results  in the Result panel to browse more rows in the result set.

The rows have the following data:

```
OrderID: 24
Status: open
OrderDate: 2003-02-06
CompanyName: Landmark Systems
CustomerContactFirstName: Joyce
CustomerContactLastName: Landers
CustomerContactPhone: (212) 333-1000
ProductID: 23
Discount: 0.05
ProductName: Widget 5
LeadTime: 1 Day
TransactionID: 30
DateRequired: 2003-02-10
DatePromised: 2003-02-10
ShipDate: 2003-02-12
SupplierID: 5
SupplierName: Good Supplies International
```

```
SupplierContactName: Justin Glory
SupplierPhoneNumber: (415) 665-8000
```

This result set provides a consolidated view of the customer-reported issue, and also contains information that can be used to find a solution to the current problem. You can contact the customer and offer a better discount. You can also contact the supplier to report the seriousness of the problem and negotiate a price reduction.

`CompositeView` is the single unified view that represents your business solution.

**Step 11** Close the `CompositeView` tab.

Optionally, you can publish each view to make it available to client applications that connect to the server through JDBC and/or ODBC. To do so, you must publish a view as a CIS database table. After you publish a view, you can re-use it as a relational database table to query further.

For further details on publishing, see the *CIS User Guide*.

## Publishing Your Views

The location to publish a view for JDBC or ODBC client applications is Data Services in your Desktop.

CIS data services are the entry points for external applications to communicate with CIS Server and the metadata. Therefore, this is where you will publish the resources that you want to make available for client applications. You will publish the views you have created so far to a CIS data service of the type CIS database, which is similar to other relational databases and which you can query just as you would query a normal database.

This section describes how to create a CIS data service of the type CIS database. For a quick tour of how to create and publish a REST service, see the tutorials in the *CIS User Guide*.

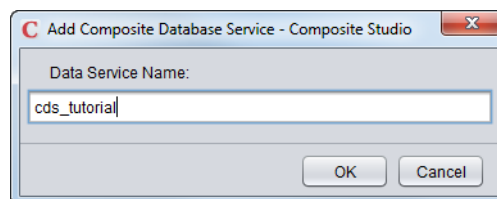
### To create a CIS data service of the type CIS database

**Step 1** Right-click *Data Services > Databases* in the resource tree.

**Step 2** Select New Composite Database Service.

The Add Data Service window prompts you to enter a name for the service.

**Step 3** Type `cds_tutorial`, as the unique name for the CIS data service you are creating.



**Step 4** Click **OK**.

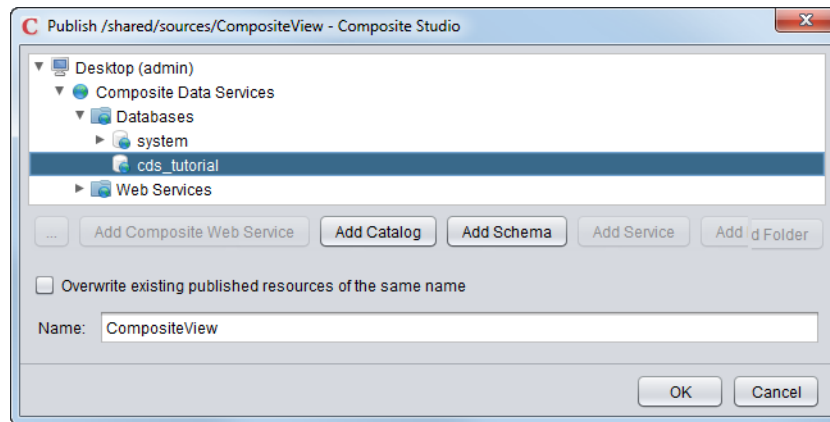
This entry is similar to a database name, and will display in the resource tree.

## Publishing Views to CIS Databases

This section describes how to publish your views to the newly created CIS database (`cds_tutorial`). The process is similar for publishing other resources.

### To publish CompositeView to a CIS database

- Step 1** Expand *Shared > sources* in the resource tree.
- Step 2** Right-click CompositeView, and select the Publish option.
- Step 3** In the Publish window, specify the location to publish the view, by navigating to *Data Services > Databases > cds\_tutorial*:



Optionally, you can add a catalog and a schema and specify the schema as the location to publish the view.

- Step 4** You can accept the default name displayed in the Name field, and click **OK**.  
The view is now published.
- Step 5** You can expand *Data Services > Databases > cds\_tutorial* in the resource tree to see the view published as CompositeView.
- Step 6** Optionally, you can publish the other views.

## What You Learned in this Tutorial

During this tutorial:

- You created three separate views.
- You learned how to introspect the underlying physical data sources for selecting only the tables you needed for your modeling, and also to construct views.
- You created views to represent the activities of a particular department (Order, Purchase, or Sales) at ALPHA Manufacturing. Each department conducts its own business without having to share information with another department so the views represent internal organizational activities.
- You modelled the final, unified view from three individual views.
- You learned how to use a view as a building block in its own capacity. Because you can use a CIS database as any relational database, you can build any number or type of views you want until a view matches your business need and publish that view as a CIS database table. This task demonstrates the reusability of a view.

- You learned how to publish the composite view (`CompositeView`). The view was based on three separate views (`ViewOrder`, `ViewSupplier` and `ViewSales`). Published views are like virtual database tables and you can query them as you would the tables in a relational database.

There are many more modeling and publishing tasks you can perform using CIS, such as transforming data, creating procedures, publishing views and procedures as CIS databases and Web services, and setting up security.

For details on security and other modeling tasks, see the *CIS User Guide*.





## CHAPTER 4

# Sample Resources

---

This chapter gives an overview of the sample resources that are available when you launch Studio.

Two relational data sources are used in this example. The inventory data source stores information about ALPHA's inventory transactions, products, purchase orders, and suppliers. The orders data source stores information about ALPHA's customers, order details, orders, and shipping methods.

The third data source is a product catalog. It has information about ALPHA's products including prices, reorder level, and lead time, all of which is stored in an XML data source named productCatalog.xml.

Topics covered in this chapter include:

- [Sample Data Sources, page 39](#)
- [Sample Transformations, page 40](#)
- [Sample Definition Set, page 40](#)
- [Sample Published View, page 41](#)
- [Sample Views, page 41](#)
- [Sample SQL Script, page 41](#)

## Sample Data Sources

Three physical data sources pre-loaded with data are in the CIS metadata repository. The data sources are two database tables and one XML file.

| Logical Names | Resource Tree Name | Data Source Type         | Table Names                                                      |
|---------------|--------------------|--------------------------|------------------------------------------------------------------|
| orders        | ds_orders          | set of relational tables | inventorytransactions<br>products<br>purchaseorders<br>suppliers |

| Logical Names      | Resource Tree Name | Data Source Type        | Table Names                                                                                                                                                                                                 |
|--------------------|--------------------|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| inventory          | ds_inventory       | set of relational table | customers<br>orderdetails<br>orders<br>shippingmethods                                                                                                                                                      |
| productCatalog.xml | ds_XML             | XML file                | The ds_XML data source has one file named productCatalog.xml. The file contains product category information in XML format. To query an XML data source, you must transform the XML data into tabular form. |

# Sample Transformations

Two sample transformations are displayed in the resource tree:

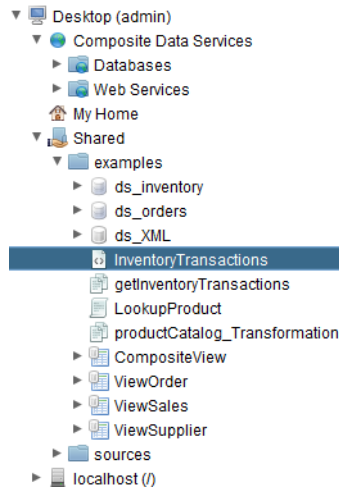
- `getInventoryTransactions` — transforms tabular data from different sources into XML form using the definition set `InventoryTransactions`.
- `productCatalog_Transformation` — transforms the XML data in `ds_XML` into tabular form.

For details on transformations, see the *CIS User Guide*.

# Sample Definition Set

An XML-type definition set, `InventoryTransactions`, has been pre-created. You can use it to create other resources such as a transformation similar to `getInventoryTransactions`.

For details on definition sets, see the *CIS User Guide*.



## Sample Published View

Resources in *Desktop > Data Services* are considered published. One sample view, CompositeView, was created during the tutorial steps. It has been made available to client applications that use JDBC or ODBC to connect to CIS Server.

When Designer projects are deployed and published on the CIS Server the associated resource definitions and services (that are Read-Only) are also displayed in the Data Services node.

For detailed information on publishing resources, see the *CIS User Guide*.

## Sample Views

The following four sample views exist in the resource tree:

| View Name     | Description                                                                                                                                                                                                                                                                                                                                                             |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CompositeView | This view provides a unified, composite view of other views, ViewOrder, ViewSales, and ViewSupplier. It examines the three separate views and joins them on one column ProductID.                                                                                                                                                                                       |
| ViewOrder     | This view provides a specific view of the ds_orders data source and retrieves details about orders and customers. It examines three tables—customers, orders, and orderdetails—from the ds_orders data source, and joins the tables on two columns, CustomerID and OrderID. Then, it selects specific columns for the result set.                                       |
| ViewSales     | This view provides a combined view of a relational data source (ds_orders) and an XML-type of file data source (productCatalog.xml). The view examines the orderdetails table from ds_inventory and the entire productCatalog_transformation, and joins them on one column, ProductID. Then, it selects specific columns for projection in the execution result set.    |
| ViewSupplier  | This view provides a specific view of the ds_inventory data source and retrieves information about inventory and suppliers. It examines three tables—inventorytransactions, purchaseorders, and suppliers—from the ds_inventory data source, and joins the tables on two columns, PurchaseOrderID and SupplierID. Then, it selects specific columns for the result set. |

These views are unpublished and not ready for external client access. For details on views, see the *CIS User Guide*.

## Sample SQL Script

A SQL script, LookupProduct, has been pre-created. You can use this script in other resources, for example the sample transformation named getInventoryTransactions.

For details on SQL scripts, see the *CIS User Guide*.





## INDEX

---

### A

adding  
    data sources [3-16](#)  
administration [1-8](#)  
alias [3-30](#)  
annotations [1-9](#)  
audience [2-3](#)

---

### B

book purpose [2-3](#)  
building views [3-27](#)  
build time [1-7](#)

---

### C

Caching panel [3-29](#)  
cardinality statistics [3-29](#)  
catalog [3-37](#)  
CIS documentation  
    searching [2-4](#)  
CIS Server [1-8](#)  
Columns panel [3-28](#)  
Composite database  
    Composite data service type [3-36](#)  
    creating [3-16](#)  
    table [3-36](#)  
Composite data services [2-13, 3-36](#)  
    creating [3-36](#)  
Composite data source [3-16](#)  
Composite Designer deployed projects [4-41](#)  
Composite resources [1-9](#)

Composite Server  
    connecting to [2-11](#)  
CompositeView [4-41](#)  
composite view [3-35](#)  
connecting [1-10](#)  
conventions in this manual [2-3](#)  
create  
    folders [3-27](#)  
    link [3-26](#)  
creating [1-10](#)  
    Composite data services [3-36](#)  
credentials [2-12](#)

---

### D

data modeling [1-8, 1-9](#)  
data services [1-9](#)  
data source  
    adding [3-15](#)  
data sources  
    adding [3-16](#)  
data source schema  
    view [3-20](#)  
Data Virtualization Platform [1-7](#)  
data virtualization server [1-8](#)  
definition  
    Composite data source [3-16](#)  
definition set [1-10, 4-40](#)  
Designer [1-8](#)  
Desktop [2-13](#)  
documentation  
    searching the CIS documentation set [2-4](#)  
documentation conventions [2-3](#)

Document Change History [2-5](#)

domain [2-12](#)

---

## E

encryption [2-12](#)

examine data sources [3-15](#)

examples [2-13](#)

executing

views [3-30](#)

execution results [3-31](#)

---

## F

federated data [1-8](#)

federation [1-7](#)

File-XML [3-22](#)

flattened data [3-24](#)

Foreign Keys panel [3-28](#)

---

## G

Grid panel [3-28](#)

---

## H

hierarchical data [1-9](#)

host [2-12](#)

HTTP clear text [2-12](#)

---

## I

Indexes panel [3-28](#)

Info panel [3-29](#)

inner join [3-29](#)

introspecting [1-10](#)

---

## J

Java procedures [1-9](#)

---

## K

Kerberos [2-12](#)

---

## L

leaf-node [2-14](#)

localhost [2-12, 3-18, 3-21](#)

logging in [2-11](#)

requirements [2-11](#)

LookupProduct [4-41](#)

---

## M

metadata repository [1-8, 3-15](#)

Modeler [2-13](#)

Model panel [3-28](#)

multi-selection [3-26](#)

My Home [2-13](#)

---

## N

Netezza [1-7](#)

---

## O

opening

resources [2-14](#)

overview [1-7](#)

---

## P

packaged applications [1-7](#)

packaged queries [1-9](#)

- parameterized queries [1-9](#)
- parent container [1-9](#)
- PDF files
  - searching across multiple [2-4](#)
- port [2-12, 3-18, 3-21](#)
- productCatalog\_transformation [4-41](#)
- productCatalog.xml [3-22](#)
- publish [3-16, 3-36](#)
- published [4-41](#)
- publishing [1-10, 4-41](#)
  - views [3-36](#)

---

## Q

- queries
  - executing [3-30](#)
  - saving [3-30](#)
- query engine [1-8](#)

---

## R

- refresh
  - resource tree [3-22](#)
- repository database [2-11](#)
- Resource Names [1-9](#)
- resources [1-9](#)
- resource tree [2-13, 3-20, 3-22](#)
  - refresh [3-22](#)
- result set
  - limiting [3-29](#)
- run time [1-7](#)

---

## S

- saving
  - views [3-30](#)
- schema [3-37](#)
- searching documentation [2-4](#)

- security [3-38](#)
- Shared [2-13](#)
- simple views [3-15](#)
- single view [1-7](#)
- SQL panel [3-28](#)
- SQL script [4-41](#)
- SQL scripts [1-9](#)
- SSL [2-12](#)
- Studio [1-8](#)
- studio
  - starting [2-11](#)

---

## T

- tabular data [1-9](#)
- Test Identity [3-29](#)
- transformation [1-8, 3-24, 3-25](#)
- transformations [1-9](#)
- transforming
  - hierarchical data [3-24](#)
  - XML [3-24](#)
- triggers [1-10](#)
- tutorial [3-15](#)

---

## U

- unified view [1-7, 3-15](#)
- unique identifier [1-9](#)

---

## V

- view
  - building [3-27](#)
- ViewOrder [4-41](#)
- views
  - building [3-27](#)
  - publishing [3-36](#)
  - reusability [3-37](#)

ViewSales [4-41](#)

ViewSupplier [4-41](#)

---

## W

Web services [1-7](#)

---

## X

XML [1-7](#)

XML data source

    adding [3-22](#)

XSLT Transformation [3-24](#)