



Enterprise Architecture Modeling
PowerDesigner® 16.5

Windows

DOCUMENT ID: DC00816-01-1650-01

LAST REVISED: January 2013

Copyright © 2013 by Sybase, Inc. All rights reserved.

This publication pertains to Sybase software and to any subsequent release until otherwise indicated in new editions or technical notes. Information in this document is subject to change without notice. The software described herein is furnished under a license agreement, and it may be used or copied only in accordance with the terms of that agreement.

Upgrades are provided only at regularly scheduled software release dates. No part of this publication may be reproduced, transmitted, or translated in any form or by any means, electronic, mechanical, manual, optical, or otherwise, without the prior written permission of Sybase, Inc.

Sybase trademarks can be viewed at the Sybase trademarks page at <http://www.sybase.com/detail?id=1011207>. Sybase and the marks listed are trademarks of Sybase, Inc. ® indicates registration in the United States of America.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries all over the world.

Java and all Java-based marks are trademarks or registered trademarks of Oracle and/or its affiliates in the U.S. and other countries.

Unicode and the Unicode Logo are registered trademarks of Unicode, Inc.

All other company and product names mentioned may be trademarks of the respective companies with which they are associated.

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

Contents

CHAPTER 1: Getting Started with Enterprise Architecture Modeling	1
Creating an EAM	3
EAM Properties	5
Creating an Enterprise Architecture Framework Project	5
Customizing your Modeling Environment	6
Setting Model Options	6
Setting EAM Display Preferences	7
Extending your Modeling Environment	7
Linking Objects with Traceability Links	8
CHAPTER 2: Business Layers Diagrams	9
Organization Charts	9
Organization Chart Objects	10
Business Communications Diagrams	10
Business Communication Diagram Objects	11
Process Maps	12
Process Map Objects	13
City Planning Diagrams	14
City Planning Diagram Objects	15
Organization Units (EAM)	15
Creating an Organization Unit	16
Organization Unit Properties	16
People (EAM)	17
Creating a Person	17
Person Properties	18
Roles (EAM)	18
Creating a Role	19

Role Properties	19
Role Associations	20
Creating a Role Association	20
Architecture Areas (EAM)	20
Creating an Architecture Area	21
Architecture Area Properties	21
Sites (EAM)	22
Creating a Site	23
Site Properties	23
Business Functions (EAM)	24
Creating a Business Function	24
Business Function Properties	24
Processes (EAM)	25
Creating a Process	26
Process Properties	26
Business Flows (EAM)	26
Creating a Business Flow	27
Business Flow Properties	27
CHAPTER 3: Application Layer Diagrams	29
Application Architecture Diagrams	29
Application Architecture Diagram Objects	30
Service-Oriented Diagrams	32
Service-Oriented Diagram Objects	34
Application and Business Services (EAM)	35
Creating an Application or Business Service	35
Application and Business Service Properties	36
Creating Business and Application Service	
Operations	37
Operation Properties	37
Systems, Applications, and Databases (EAM)	38
Creating a System, Application, or Database	39
System, Application, and Database Properties	39
Components (EAM)	41

Creating a Component	41
Component Properties	42
Forms, Documents, and Reports (EAM)	42
Creating a Form, Document, or Report	43
Form, Document, and Report Properties	43
Creating Document Data	44
Data Properties	44
ETL Jobs (EAM)	45
Creating an ETL Job	45
ETL Job Properties	46
Application Links (EAM)	46
Creating an Application Link	47
Application Link Properties	47
Contracts (EAM)	48
Creating a Contract	48
Contract Properties	48
CHAPTER 4: Technology Layer Diagrams	51
Technology Infrastructure Diagram Objects	52
Networks (EAM)	53
Creating a Network	53
Network Properties	54
Servers, Workstations, Mobile Devices, and Network	
Nodes (EAM)	54
Creating a Software or Hardware Server, Workstation,	
Mobile Device, or Network Node	55
Software and Hardware Server, Workstation, Mobile	
Device, and Network Node Properties	55
Modeling Cluster Servers and Virtual Machines	56
Deployment Instances (EAM)	57
Creating a Deployment Instance	58
Deployment Instance Properties	58
Infrastructure Links (EAM)	59
Creating an Infrastructure Link	59

Infrastructure Link Properties	59
CHAPTER 5: Sybase IQ Reference Architecture	
Model	61
Creating a Reference Architecture Model	61
Creating Reference Architectures	61
Reference Architecture Properties	67
Database Server Properties	68
IQ Server Properties	69
SAN Properties	70
IQ Store Properties	71
Query Type Properties	71
Reference Architecture Model Properties	72
Logical Server Properties	73
Comparing Reference Architectures	73
Modifying and Regenerating Reference Architectures	74
Generating Reports from a Reference Architecture	
Model	75
 CHAPTER 6: Hardware Profile Library	 77
Importing Profiles from the SPEC Website	77
Hardware Profile Properties	78
Hardware Profile Library Properties	79
Specifying Additional Query Types	79
 CHAPTER 7: Checking an EAM	 81
Package Checks	81
Application, Application Service, Component and	
Database Checks	82
Application Link Checks	83
Contract Checks	84
Deployment Instance Checks	85
Form Checks	86

Hardware Server, Mobile Device, Network Node and Workstation Checks	86
Network Checks	87
Organization Unit Checks	88
Person Checks	89
Software Server Checks	90
CHAPTER 8: Importing to and Generating and Exporting from an EAM	91
Importing Visio Diagrams into PowerDesigner	91
Exporting and Importing Objects to and from Other Models	92
Exporting Objects from an EAM to Another Model	93
Importing Objects from Another Model into an EAM	94
Generating an EAM from an EAM	95
Index	97

Contents

Getting Started with Enterprise Architecture Modeling

An *enterprise architecture model (EAM)* helps you analyze and document your organization and its business functions, along with the applications and systems that support them and the physical architecture on which they are implemented.

Enterprise architecture modeling helps you to analyze and document the architecture of your organization. Such an analysis, can be required:

- As an end in itself as part of good *corporate governance*
- Where you are required to document your system's *compliance* with standards or regulations
- In the following kinds of *change management* situations:
 - Rationalization – where cost savings and efficiencies are sought
 - A change in technology – where your system will be upgraded
 - An acquisition or merger – where two disparate systems must be made to work together

The diagrams in the PowerDesigner® enterprise architecture model are divided into three layers as follows:

- *Business Layer* - helps you to analyze your organization in various ways:
 - *Organization Charts* – for documenting groups and people
 - *Business Communication Diagram* – for documenting sites and analyzing the interactions between various parts of your organization
 - *Process Map* – for documenting functions
 - *City Planning Diagram* - for providing a big picture, global view of the organization, and for descending to the system and database level, to provide links into the next, application layer.
- *Application Layer* - descends into the applications, databases, and other systems that must implement the business functions identified in the business layer. The two diagrams in the application layer allow you to identify what applications you have, and how they interact with each other, in two different ways:
 - *Application Architecture Diagram* - as traditional discrete applications
 - *Service Oriented Diagram* - in terms of SOA services
- *Technology Layer* - models the hardware on which the software is deployed. If the business layer allows you to define what it is your organization does, and the application layer identifies the software that implements these functions, the technology layer documents the physical equipment on which the software will run. This layer has a single diagram:

CHAPTER 1: Getting Started with Enterprise Architecture Modeling

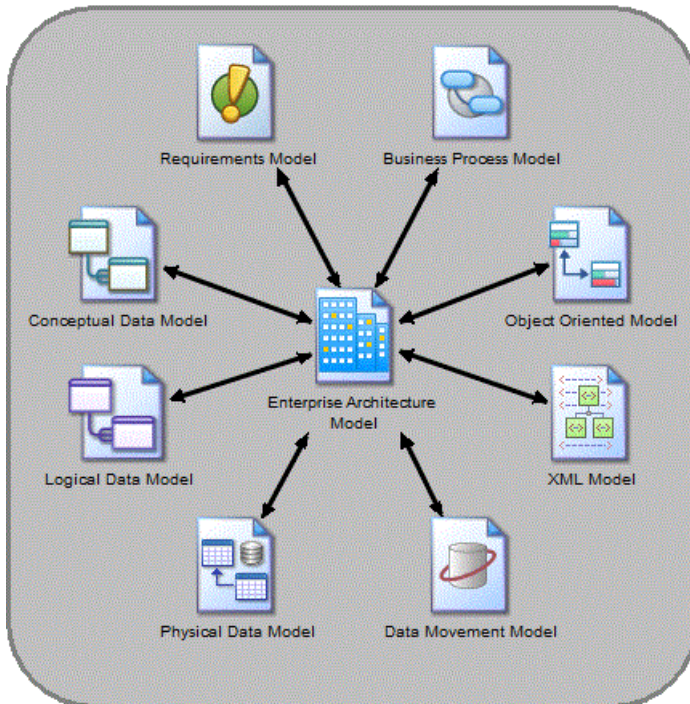
- *Technology Infrastructure Diagram* – for documenting your existing infrastructure, your networks, server clusters, firewalls, and client workstations, and the infrastructure that you want to put in place.

Extending Your Analysis into Other Modules

The enterprise architecture model is intended to give you the big picture of your organization, and to provide means to decompose your functions, processes, and systems to a certain level of detail. However, when it comes time to model the implementation of databases, web services, or OO components, you will want to do this in the PowerDesigner module designed for the task.

The EAM contains wizards that allow you to import objects from and export objects to other PowerDesigner modules. Objects exported or imported remain linked to the original objects, in order to keep all your models synchronized and to allow you to perform cross-module impact analysis to identify the technical impact of changes on your enterprise architecture.

The following diagram shows how you can enrich the analysis of your architecture by linking your EA objects with lower-level objects in other types of PowerDesigner models:



Using Enterprise Architecture Frameworks

PowerDesigner projects allow you to easily combine multiple models and view the connections between them. You can follow an enterprise architecture framework such as

FEAF by creating a project that combines your EAM diagrams with those of a Physical Data Model and other PowerDesigner modules.

For more information, see *Creating an Enterprise Architecture Framework Project* on page 5.

Suggested Bibliography

- O'Rourke, Fishman, Selkow, *Enterprise Architecture: Using the Zachman Framework*, Thompson Course Technology, 2003, 716 pages, ISBN 0-619-06446-3.
- David C. Hay, Morgan Kaufmann, *Data Model Patterns: A Metadata Map*, 2006, 406 pages, ISBN 0-12-088798-3.

The following online resources provide an introduction to enterprise architecture concepts:

- Wikipedia on enterprise architecture and service-oriented modeling:
 - http://en.wikipedia.org/wiki/Enterprise_architecture
 - http://en.wikipedia.org/wiki/Service-Oriented_Modeling
- The Zachman Institute for Framework Advancement (ZIFA):
 - <http://www.zifa.com/>
- The Federal Enterprise Architecture Framework (FEAF):
 - <http://www.cio.gov/Documents/bpeaguide.pdf>
- The Department of Defense Architecture Framework (DODAF):
 - http://www.defenselink.mil/cio-nii/docs/DoDAF_Volume_I.pdf

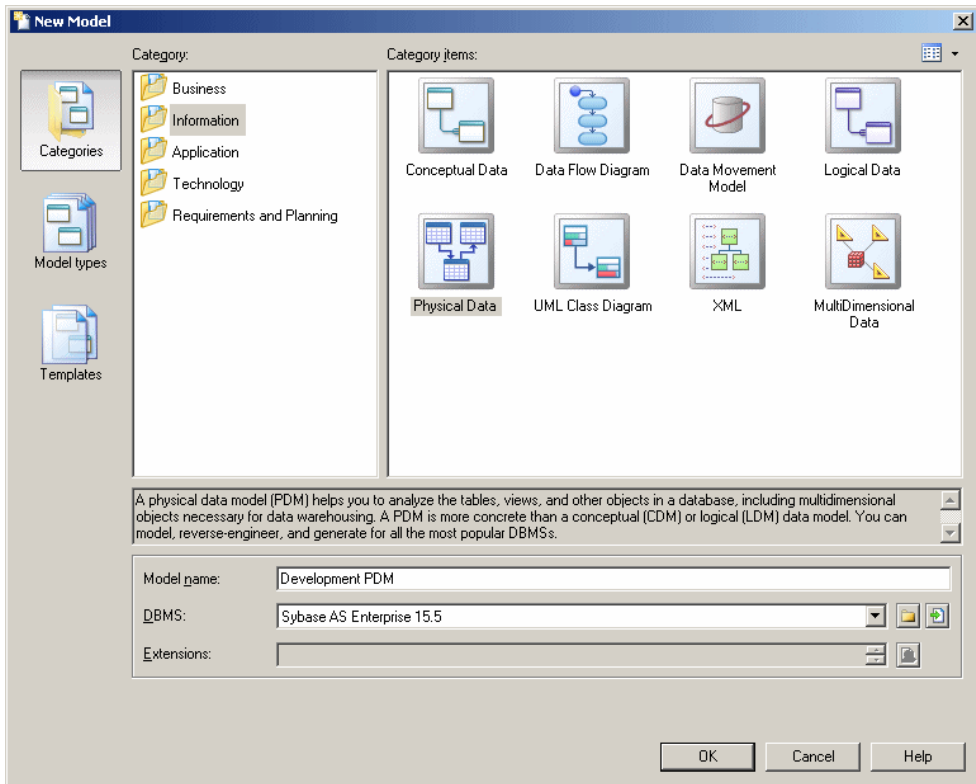
Creating an EAM

You create a new enterprise architecture model by selecting **File > New Model**.

Note: In addition to creating an EAM from scratch with the following procedure, you can also create a model by importing a Visio file (see *Importing Visio Diagrams into PowerDesigner* on page 91).

The New Model dialog is highly configurable, and your administrator may hide options that are not relevant for your work or provide templates or predefined models to guide you through model creation. When you open the dialog, one or more of the following buttons will be available on the left hand side:

- **Categories** - which provides a set of predefined models and diagrams sorted in a configurable category structure.
- **Model types** - which provides the classic list of PowerDesigner model types and diagrams.
- **Template files** - which provides a set of model templates sorted by model type.



1. Select **File > New Model** to open the New Model dialog.
2. Click a button, and then select a category or model type (**Enterprise Architecture Model**) in the left-hand pane.
3. Select an item in the right-hand pane. Depending on how your New Model dialog is configured, these items may be first diagrams or templates on which to base the creation of your model.
Use the **Views** tool on the upper right hand side of the dialog to control the display of the items.
4. Enter a model name. The code of the model, which is used for script or code generation, is derived from this name using the model naming conventions.
5. [optional] Click the **Select Extensions** button and attach one or more extensions to your model.
6. Click **OK** to create and open the enterprise architecture model .

Note: Sample EAMs are available in the Example Directory.

EAM Properties

You open the model property sheet by right-clicking the model in the Browser and selecting **Properties**.

Each enterprise architecture model has the following model properties:

Property	Description
Name/Code/Comment	Identify the model. The name should clearly convey the model's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the model. By default the code is auto-generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Filename	Specifies the location of the model file. This box is empty if the model has never been saved.
Author	Specifies the author of the model. If you enter nothing, the Author field in diagram title boxes displays the user name from the model property sheet Version Info tab. If you enter a space, the Author field displays nothing.
Version	Specifies the version of the model. You can use this box to display the repository version or a user defined version of the model. This parameter is defined in the display preferences of the Title node.
Default diagram	Specifies the diagram displayed by default when you open the model.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.
















Creating an Enterprise Architecture Framework Project

PowerDesigner provides support for some of the most popular enterprise architecture frameworks in the form of project templates.

1. Select **File > New Project** to open the New Project dialog.
2. In the Project type pane, expand the EA Frameworks node and select one of the available frameworks.
3. Specify a name for the project and a location to save it, and then click OK to create the project.

The project opens to the framework matrix, from which you can create models, diagrams, lists, and other items to complete the requirements of the framework:

CHAPTER 1: Getting Started with Enterprise Architecture Modeling

	What	How	Where
Planner's view	List of Business Objects 	List of business processes 	List of Business locations 
Owner's view	Semantic model 	Business process Model 	Business logistic system 
Designer's view	Logical data model 	Application architecture 	Deployment architecture 
Builder's view	Physical data model 	System design 	Technology architecture 
Subcontractor's view	Data directory 	Programs 	Network architecture 

For detailed information about working with projects, see *Core Features Guide > Modeling with PowerDesigner > Projects and Frameworks*.

Customizing your Modeling Environment

The PowerDesigner enterprise architecture model provides various means for customizing and controlling your modeling environment.

Setting Model Options

You can set EAM model options by selecting **Tools > Model Options** or right-clicking the diagram background and selecting **Model Options**.

You can set the following options on the Model Settings page:

Option	Description
Name/Code case sensitive	Specifies that the names and codes for all objects are case sensitive, allowing you to have two objects with identical names or codes but different cases in the same model. If you change case sensitivity during the design process, we recommend that you check your model to verify that your model does not contain any duplicate objects.
Enable links to requirements	Displays a Requirements tab in the property sheet of every object in the model, which allows you to attach requirements to objects (see <i>Requirements Modeling</i>).
External Short-cut Properties	Specifies the properties that are stored for external shortcuts to objects in other models for display in property sheets and on symbols. By default, All properties appear, but you can select to display only Name/Code to reduce the size of your model. Note: This option only controls properties of external shortcuts to models of the same type (PDM to PDM, EAM to EAM, etc). External shortcuts to objects in other types of model can show only the basic shortcut properties.

For information about controlling the naming conventions of your models, see *Core Features Guide > Modeling with PowerDesigner > Objects > Naming Conventions*.

Setting EAM Display Preferences

PowerDesigner display preferences allow you to customize the format of object symbols, and the information that is displayed on them. To set enterprise architecture model display preferences, select **Tools > Display Preferences** or right-click the diagram background and select **Display Preferences** from the contextual menu.

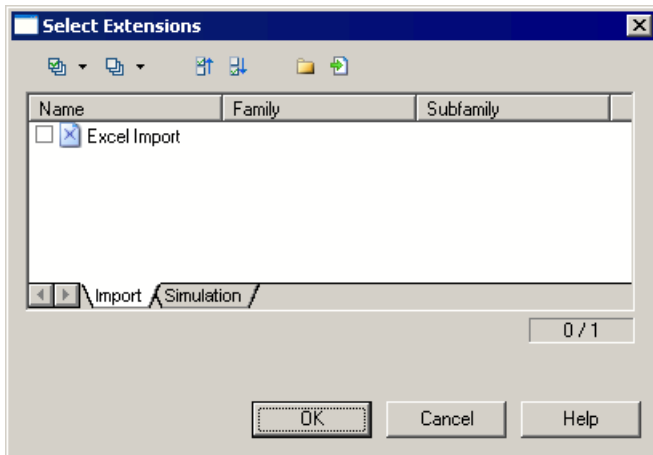
For detailed information about customizing and controlling the attributes and collections displayed on object symbols, see *Core Features Guide > Modeling with PowerDesigner > Diagrams, Matrices, and Symbols > Display Preferences*.

Extending your Modeling Environment

You can customize and extend PowerDesigner metaclasses, parameters, and file generation with extensions, which can be stored as part of your model or in separate extension files (*.xem) for reuse with other models.

To access extension defined in a *.xem file, simply attach the file to your model. You can do this when creating a new model by clicking the **Select Extensions** button at the bottom of the New Model dialog, or at any time by selecting **Model > Extensions** to open the List of Extensions and clicking the **Attach an Extension** tool.

In each case, you arrive at the Select Extensions dialog, which lists the extensions available, sorted on sub-tabs appropriate to the type of model you are working with:



To get started extending objects, see *Core Features Guide > Modeling with PowerDesigner > Objects > Extending Objects*. For detailed information about working with extensions, see *Customizing and Extending PowerDesigner > Extension Files*.

Linking Objects with Traceability Links

You can create traceability links to show any kind of relationship between two model objects (including between objects in different models) via the **Traceability Links** tab of the object's property sheet. These links are used for documentation purposes only, and are not interpreted or checked by PowerDesigner.

For more information about traceability links, see *Core Features Guide > Linking and Synchronizing Models > Getting Started with Linking and Syncing > Creating Traceability Links*.

The diagrams in the business layer of the enterprise architecture model help you to analyze your organization in terms of business units, processes, and functions:

- Organization Charts – provide graphical representations of your organization's hierarchies (see *Organization Charts* on page 9).
- Business Communication Diagrams – provide a consistent framework for interactions between components of the organization, and with customers, suppliers and partners (see *Business Communications Diagrams* on page 10).
- Process Maps – provide a view of how high-level processes are associated with business functions (see *Process Maps* on page 12).
- City Planning Diagrams – provide a "big picture" view of the organization broken down into architectural areas (see *City Planning Diagrams* on page 14).

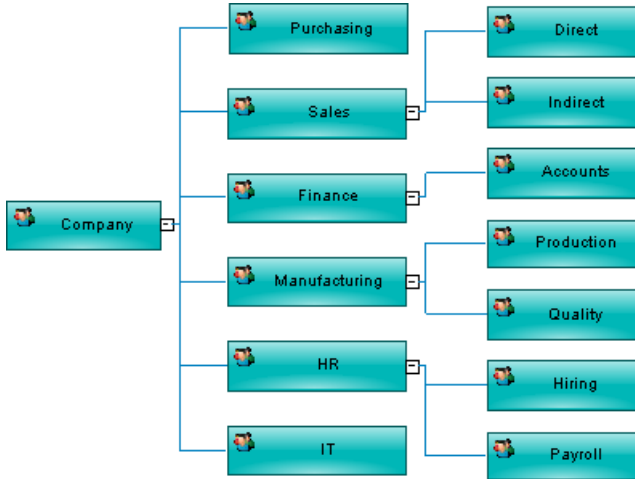
Organization Charts

An *organization chart* provides a graphical view of your organization as a tree structure, and helps you analyze and display the relationships between organization units (divisions, groups, teams, etc), individuals, and roles.

Note: To create an organization chart in an existing EAM, right-click the model in the Browser and select **New > Organization Chart Diagram**. To create a new model, select **File > New Model**, choose Enterprise Architecture Model as the model type and **Organization Chart Diagram** as the first diagram, and then click **OK**.

In the following example, the company is analyzed into its top-level departments and the major groups within them:

CHAPTER 2: Business Layers Diagrams



Organization Chart Objects

YPowerDesigner supports all the objects necessary to build organization charts.

Object	Tool	Symbol	Description
Organization unit			A group, department, function, or other collection of people or organization units. See <i>Organization Units (EAM)</i> on page 15.
Person			An individual. See <i>People (EAM)</i> on page 17.
Hierarchy link			A link between organization units or people. See <i>Roles (EAM)</i> on page 18.

Note: Most EAM objects can be displayed as icons or as boxes that can display additional properties and sub-objects. You can toggle between the Icon and Detail modes via the contextual menu, or by pressing **Ctrl+Q**. You can control the default mode, and the information displayed in each mode through the options available by selecting **Tools > Display Preferences**.

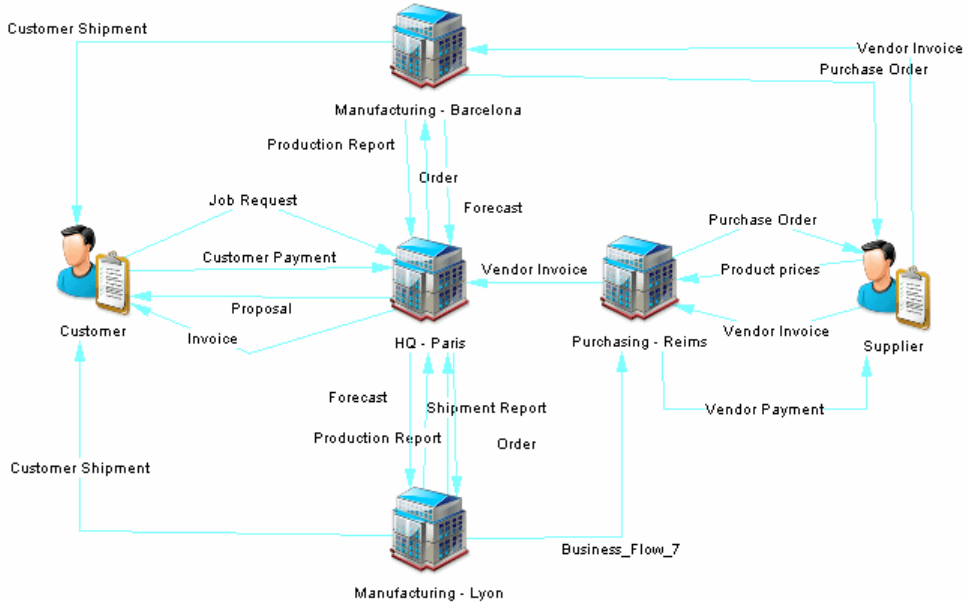
Business Communications Diagrams

A *business communication diagram* provides a graphical view of your organization, and helps you analyze, the relationships, flows, and other connections between business functions, organization units, roles, and sites.

Note: To create a business communication diagram in an existing EAM, right-click the model in the Browser and select **New > Business Communication Diagram**. To create a new model,

select **File > New Model**, choose Enterprise Architecture Model as the model type and **Business Communication Diagram** as the first diagram, and then click **OK**.









In the following example, communications between the company, its customers and suppliers are analyzed, as well as those between the different company sites:



Business Communication Diagram Objects

PowerDesigner supports all the objects necessary to build business communication diagrams.

Object	Tool	Symbol	Description
Architecture Area			See <i>Architecture Areas (EAM)</i> on page 20.
Business Function			An aggregation of processes and/or sub-functions. See <i>Business Functions (EAM)</i> on page 24.
Business flow			A link between elements in a business layer diagram. See <i>Business Flows (EAM)</i> on page 26.
Process			An ordering of tasks. See <i>Processes (EAM)</i> on page 25.

Object	Tool	Symbol	Description
Site			A physical location. See <i>Sites (EAM)</i> on page 22.
Organization unit			A group, department, function, or other collection of people or organization units. See <i>Organization Units (EAM)</i> on page 15.
Person			An individual. See <i>People (EAM)</i> on page 17.
Role			A set of responsibilities. See <i>Roles (EAM)</i> on page 18.

Note: Most EAM objects can be displayed as icons or as boxes that can display additional properties and sub-objects. You can toggle between the Icon and Detail modes via the contextual menu, or by pressing **Ctrl+Q**. You can control the default mode, and the information displayed in each mode through the options available by selecting **Tools > Display Preferences**.

Process Maps

A *process map* provides a graphical view of your business architecture, and helps you identify your business functions and high-level processes, independent of the people and business units who fulfill them.







Note: To create a process map in an existing EAM, right-click the model in the Browser and select **New > Process Map**. To create a new model, select **File > New Model**, choose Enterprise Architecture Model as the model type and **Process Map** as the first diagram, and then click **OK**.

In the following example, the major processes within the Manufacturing, Marketing, and Sales business functions are displayed:



Process Map Objects

PowerDesigner supports all the objects necessary to build process maps.

Object	Tool	Symbol	Description
Architecture Area			An abstract container object. See <i>Architecture Areas (EAM)</i> on page 20.
Business Function			An aggregation of processes and/or sub-functions. See <i>Business Functions (EAM)</i> on page 24.
Process			An ordering of tasks. See <i>Processes (EAM)</i> on page 25.

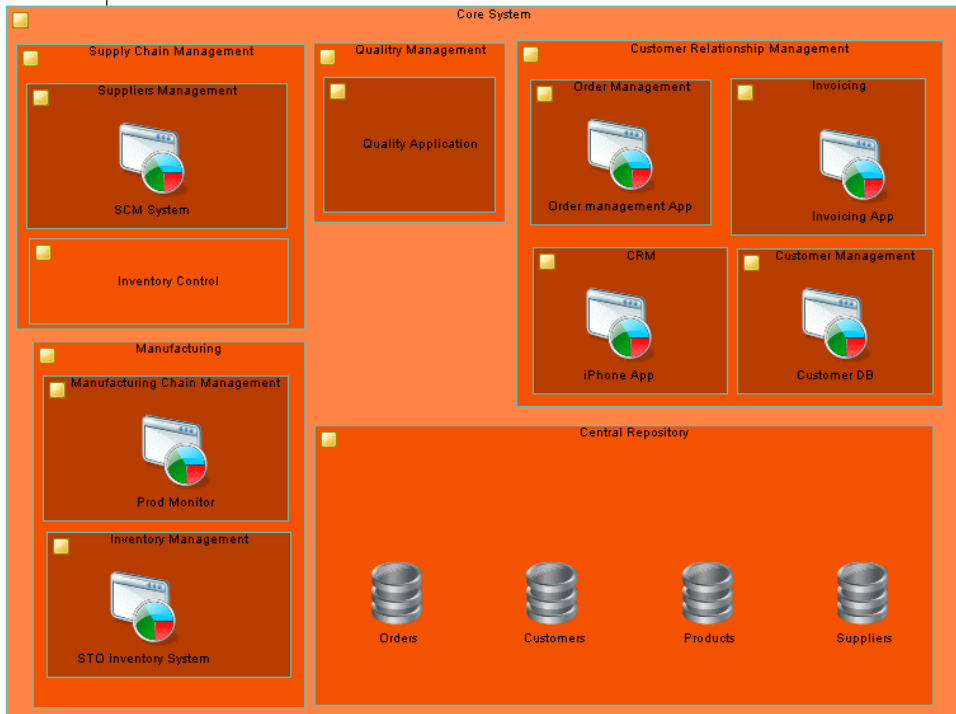
Note: Most EAM objects can be displayed as icons or as boxes that can display additional properties and sub-objects. You can toggle between the Icon and Detail modes via the contextual menu, or by pressing **Ctrl+Q**. You can control the default mode, and the information displayed in each mode through the options available by selecting **Tools > Display Preferences**.

City Planning Diagrams

A *city planning diagram* provides a graphical view of the big picture of your enterprise architecture, using the metaphor of planning the infrastructure of a city to represent the organization of systems, applications, etc into architectural areas.











Note: To create a city planning diagram in an existing EAM, right-click the model in the Browser and select **New > City Planning Diagram**. To create a new model, select **File > New Model**, choose Enterprise Architecture Model as the model type and **City Planning Diagram** as the first diagram, and then click **OK**.

In the following example, the core system is broken up into five major sub-systems, which are in turn sub-divided into their major functions. Major applications and databases are also highlighted.



City Planning Diagram Objects

PowerDesigner supports all the objects necessary to build city planning diagrams.

Object	Tool	Symbol	Description
Architecture Area			An abstract container object. See <i>Architecture Areas (EAM)</i> on page 20.
System			A group of applications, services or sub-systems. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Application			A computer program. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Database			A database. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Business Function			An aggregation of processes and/or sub-functions. See <i>Business Functions (EAM)</i> on page 24.

Note: Most EAM objects can be displayed as icons or as boxes that can display additional properties and sub-objects. You can toggle between the Icon and Detail modes via the contextual menu, or by pressing **Ctrl+Q**. You can control the default mode, and the information displayed in each mode through the options available by selecting **Tools > Display Preferences**.

Organization Units (EAM)

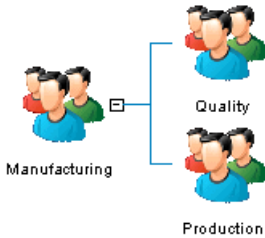
An *organization unit* represents a group of people or other organization units.

An organization unit can be created in the following diagrams:

- Organization chart (see *Organization Charts* on page 9)
- Business communication diagram (see *Business Communications Diagrams* on page 10)
- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, the Manufacturing organization unit contains the Quality and Production sub-units:



Creating an Organization Unit

You can create an organization unit from the Toolbox, Browser, or **Model** menu.

- Use the **Organization Unit** tool in the Toolbox.
- Select **Model > Organization Units** to access the List of Organization Units, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Organization Unit**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Organization Unit Properties

To view or edit an organization unit's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Manager	Specifies the manager of the organization unit. Select a user or role from the list.
Telephone	Specifies the telephone number of the organization unit.
Email	Specifies the email address of the organization unit.
Web site	Specifies the url for the web site of the organization unit.

Property	Description
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Sub-Organizations – lists the organization units that belong to this organization unit.
- People – lists the people (see *People (EAM)* on page 17) that belong to this organization unit.
- Roles – lists the roles (see *Roles (EAM)* on page 18) that this organization unit fulfills.
- Sites – lists the sites (see *Sites (EAM)* on page 22) where this organization unit is located.

People (EAM)

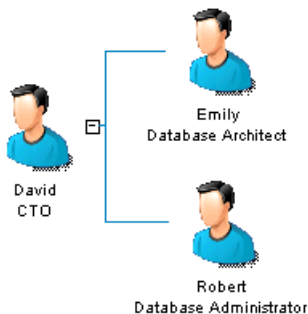
A *person* represents an individual who holds a position within an organization.

A person can be created in the following diagrams:

- Organization chart (see *Organization Charts* on page 9)
- Business communication diagram (see *Business Communications Diagrams* on page 10)
- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, David is the manager of Emily and Robert:



Creating a Person

You can create a person from the Toolbox, Browser, or **Model** menu.

- Use the **Person** tool in the Toolbox.
- Select **Model > People** to access the List of People, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Person**.

CHAPTER 2: Business Layers Diagrams

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Person Properties

To view or edit a person's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Manager	Specifies the manager of the person. Select a person from the list or click the New tool to the right of the field to create a new one.
Site	Specifies the site where this person is located. Select a site from the list or click the New tool to the right of the field to create a new one.
Job title	Specifies the job title of the person.
Telephone	Specifies the telephone number of the person.
Email	Specifies the email address of the person.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Roles – lists the roles (see *Roles (EAM)* on page 18) that this person fulfills.

Roles (EAM)

A *role* is set of responsibilities that are fulfilled by a person or an organization unit.

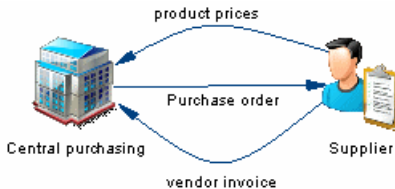
For more information, see *People (EAM)* on page 17 or *Organization Units (EAM)* on page 15.

A role can be created in the following diagrams:

- Organization chart (see *Organization Charts* on page 9)
- Business communication diagram (see *Business Communications Diagrams* on page 10)
- Service oriented diagram (see *Service-Oriented Diagrams* on page 32)
- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, the Supplier role interacts with the Central Purchasing site:



Creating a Role

You can create a role from the Toolbox, Browser, or **Model** menu.

- Select **Model > Roles** to access the List of Roles, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Role**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Role Properties

To view or edit a role's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.

Property	Description
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

Role Associations

Role associations are used to define links between objects and people, organization units, and roles.

For example:

- A person may be defined as the administrator of a hardware server and the manager of a site
- An organization unit may be identified as being responsible for the Sales business function
- A role may be designated as the contact for the Madrid site

Creating a Role Association

You can create role associations from the property sheet of the object to which you want to connect a person, organization unit, or role.

1. Open the property sheet of the object to which you want to connect a person, organization unit, or role, and click the Role Associations tab.
2. Click the Add Objects tool to open a selection box listing all the roles, people, and organization units available in the model.
3. Select the roles, people, and organization units that you want to associate with the object and click OK.
4. [optional] On the Role Associations tab, select or enter a type for each added role association.
5. Click OK to return to the diagram.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Architecture Areas (EAM)

An *architecture area* is an abstract object that can contain and group together other objects.

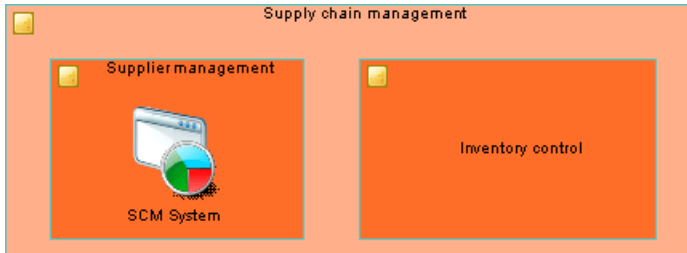
An architecture area can be created in the following diagrams:

- Business communication diagram (see *Business Communications Diagrams* on page 10)
- Process map (see *Process Maps* on page 12)
- City planning diagram (see *City Planning Diagrams* on page 14)
- Application architecture diagram (see *Application Architecture Diagrams* on page 29)

- Service oriented diagram (see *Service-Oriented Diagrams* on page 32)
- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, the Supply chain management area contains the Supplier management and Inventory control areas:



Creating an Architecture Area

You can create an architecture area from the Toolbox, Browser, or **Model** menu.

- Use the **Architecture Area** tool in the Toolbox.
- Select **Model > Architecture Areas** to access the List of Architecture Areas, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Architecture Area**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Architecture Area Properties

To view or edit an architecture area's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.

Property	Description
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Role Associations - lists the organization units, people, and roles associated with the area, and the type of role they play in relation to it (see *Role Associations* on page 20).
- Attached Objects – lists the objects that are associated with the area.
- Sub-Areas – lists the architecture areas contained within the area.

Sites (EAM)

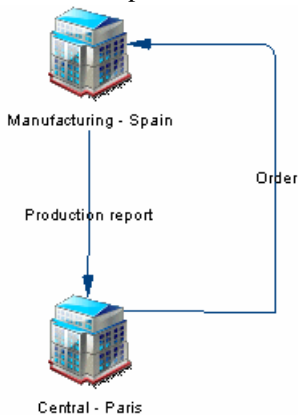
A *site* is a physical location that can contain and group people, organizations and other objects.

A site can be created in the following diagrams:

- Business communication diagram (see *Business Communications Diagrams* on page 10)
- Application architecture diagram (see *Application Architecture Diagrams* on page 29)
- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, the Central and Manufacturing sites interact:



Creating a Site

You can create a site from the Toolbox, Browser, or **Model** menu.

- Use the **Site** tool in the Toolbox.
- Select **Model > Sites** to access the List of Sites, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Site**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Site Properties

To view or edit a site's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Address	Specifies the street address of the site.
Zip code	Specifies the zip code of the site.
City	Specifies the city where the site is located.
Country	Specifies the country where the site is located.
Phone	Specifies the telephone number of the site.
Population	Specifies the number of people at the site.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- **Role Associations** – lists the organization units, people, and roles associated with the site, and the type of role they play in relation to it (see *Role Associations* on page 20).

- Sub-Sites – lists the sites contained within the site.

Business Functions (EAM)

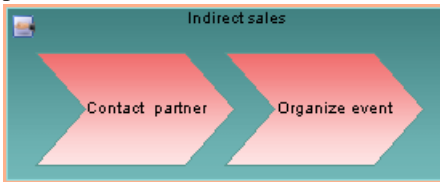
A *business function* is an aggregation of related processes and sub-functions.

A business function can be created in the following diagrams:

- Business communication diagram (see *Business Communications Diagrams* on page 10)
- Process map (see *Process Maps* on page 12)
- City planning diagram (see *City Planning Diagrams* on page 14)

Example

In this example, the Indirect sales function contains the Contact partner and Organize event processes:



Creating a Business Function

You can create a business function from the Toolbox, Browser, or **Model** menu.

- Use the **Business Function** tool in the Toolbox.
- Select **Model > Business Functions** to access the List of Business Functions, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Business Function**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Business Function Properties

To view or edit a business function's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Role Associations – lists the organization units, people, and roles associated with the function, and the type of role they play in relation to it (see *Role Associations* on page 20).
- Sub-Functions – lists the business functions contained within the function.
- Processes – lists the processes (see *Processes (EAM)* on page 25) associated with the function.
- Sites – lists the sites (see *Sites (EAM)* on page 22) associated with the function.

Processes (EAM)

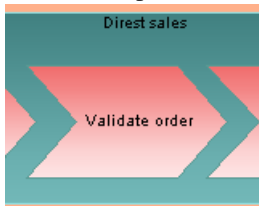
A *process* is an ordering of tasks or activities that may include manual and/or automated operations. Processes can be triggered by events, by initiatives, or by conditions.

A process can be created in the following diagrams:

- Business communication diagram (see *Business Communications Diagrams* on page 10)
- Process map (see *Process Maps* on page 12)
- Service oriented diagram (see *Service-Oriented Diagrams* on page 32)

Example

In this example, the Validate order process forms part of the Direct sales function:



Creating a Process

You can create a process from the Toolbox, Browser, or **Model** menu.

- Use the **Process** tool in the Toolbox.
- Select **Model > Processes** to access the List of Processes, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Process**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Process Properties

To view or edit a process's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Reused process	Specifies the name of the reused process.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Role Associations – lists the organization units, people, and roles associated with the process, and the type of role they play in relation to it (see *Role Associations* on page 20).

Business Flows (EAM)

A *business flow* is an oriented link that is used to connect objects and can contain documents.

A business flow can be created in the following diagrams:

- Business communication diagram (see *Business Communications Diagrams* on page 10)

Example

In this example the Price request flow goes from the Purchasing function to the Supplier role:



Creating a Business Flow

You can create a business flow from the Toolbox, Browser, or **Model** menu.

- Use the **Business Flow** tool in the Toolbox.
- Select **Model > Business Flows** to access the List of Business Flows, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Business Flow**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Business Flow Properties

To view or edit a business flow's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	<p>Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.</p> <p>Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.</p>
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.

Property	Description
First object	Specifies the object at the origin of the link.
Second object	Specifies the object at the end of the link.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Documents – lists the documents associated with the flow (see *Forms, Documents, and Reports (EAM)* on page 42).
- Condition – lets you specify a condition to be evaluated to determine if the flow will be triggered, as well as a short alias for the condition, which can be displayed on the flow symbol.

Application layer diagrams allow you to model the architecture of your systems in order to identify your applications and their main components, and to analyze their interactions and how they implement business process and functions:

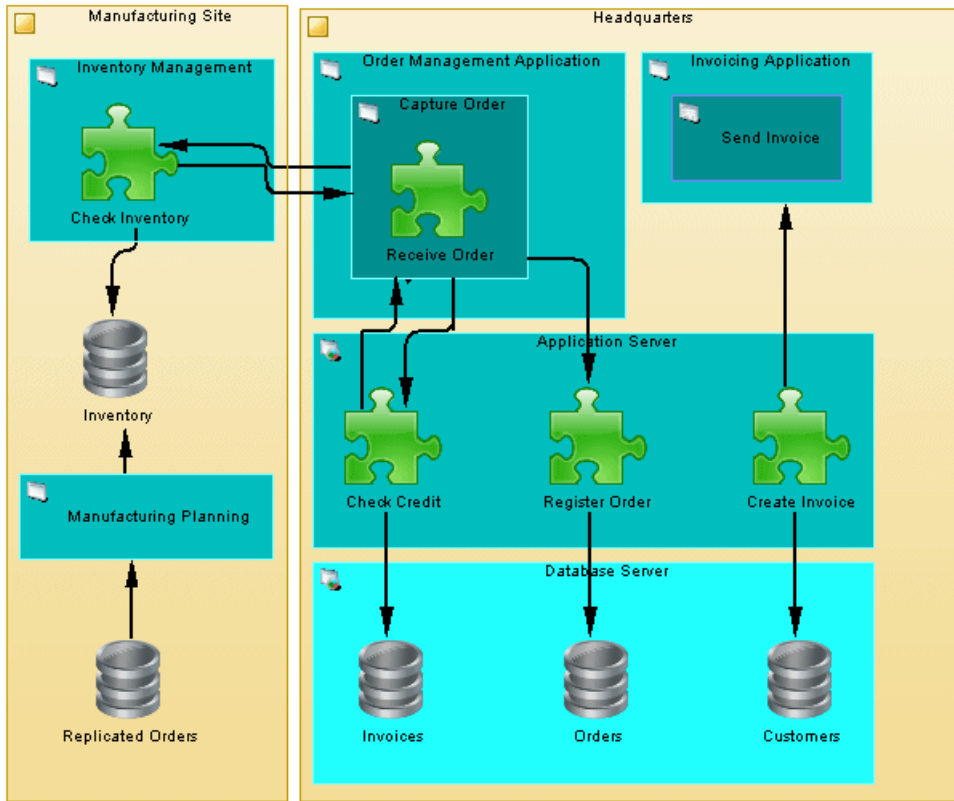
- Application Architecture Diagrams – organize applications as discrete, interconnected units (see *Application Architecture Diagrams* on page 29).
- Service Oriented Diagrams – organize applications in terms of SOA layers (see *Service-Oriented Diagrams* on page 32).

Application Architecture Diagrams

An *application architecture diagram* provides a high-level graphical view of the application architecture, and helps you identify applications, sub-applications, components, databases, services, etc, and their interactions.

Note: To create a application architecture diagram in an existing EAM, right-click the model in the Browser and select **New > Application Architecture Diagram**. To create a new model, select **File > New Model**, choose Enterprise Architecture Model as the model type and **Application Architecture Diagram** as the first diagram, and then click **OK**.



















In the following example, interactions between the major order processing and inventory systems at the company's headquarters and manufacturing site are displayed:



Application Architecture Diagram Objects

PowerDesigner supports all the objects necessary to build application architecture diagrams.

Object	Tool	Symbol	Description
Architecture Area			An abstract object for grouping other objects. See <i>Architecture Areas (EAM)</i> on page 20.
Site			A physical location. See <i>Sites (EAM)</i> on page 22.
Application Service			An externally visible unit of functionality. See <i>Application and Business Services (EAM)</i> on page 35.

Object	Tool	Symbol	Description
System			A packaged application. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Application			A computer program. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Database			A database. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Component			A replaceable part of an application. See <i>Components (EAM)</i> on page 41.
Form			A UI component. See <i>Forms, Documents, and Reports (EAM)</i> on page 42.
Document			Any conceptual document. See <i>Forms, Documents, and Reports (EAM)</i> on page 42.
Report Document			A report. See <i>Forms, Documents, and Reports (EAM)</i> on page 42.
ETL Job			An ETL job. See <i>ETL Jobs (EAM)</i> on page 45.
Application Link			A link between elements in an application layer diagram. See <i>Application Links (EAM)</i> on page 46.

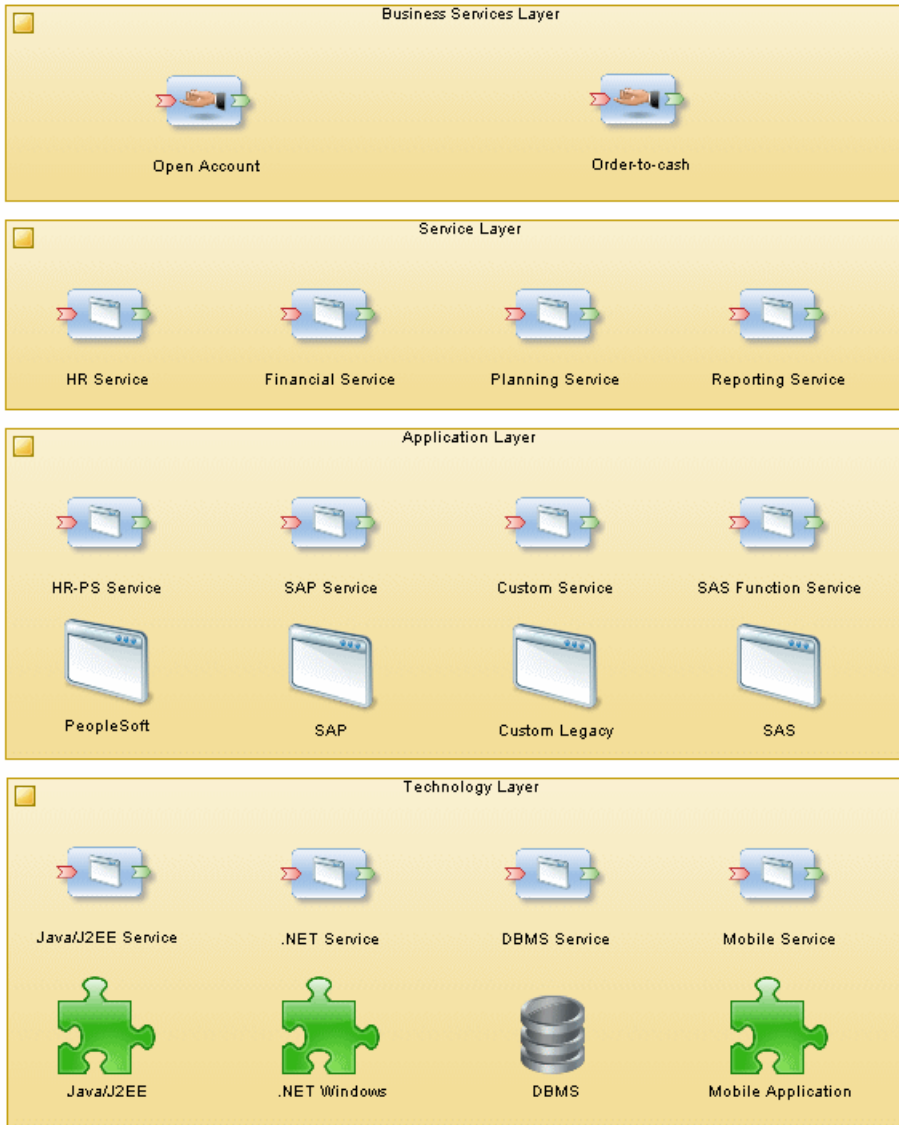
Note: Most EAM objects can be displayed as icons or as boxes that can display additional properties and sub-objects. You can toggle between the Icon and Detail modes via the contextual menu, or by pressing **Ctrl+Q**. You can control the default mode, and the information displayed in each mode through the options available by selecting **Tools > Display Preferences**.

Service-Oriented Diagrams

A *service-oriented diagram* provides a graphical view of your business and application services and the relationships between them, and helps you associate applications and other application layer objects with business services and processes to assist with SOA design.























Note: To create a service-oriented diagram in an existing EAM, right-click the model in the Browser and select **New > Service-Oriented diagram Diagram**. To create a new model, select **File > New Model**, choose Enterprise Architecture Model as the model type and **Service-Oriented Diagram** as the first diagram, and then click **OK**.

In the following example, the Open Account and Order-to-cash business services are shown in relation to the application services, applications, and technologies that implement them:



Service-Oriented Diagram Objects

PowerDesigner supports all the objects necessary to build service-oriented diagrams.

Object	Tool	Symbol	Description
Architecture Area			An abstract object for grouping other objects. See <i>Architecture Areas (EAM)</i> on page 20.
Role			A set of responsibilities. See <i>Roles (EAM)</i> on page 18.
Process			An ordering of tasks. See <i>Processes (EAM)</i> on page 25.
Application Service			An externally visible unit of functionality. See <i>Application and Business Services (EAM)</i> on page 35.
Business Service			A service offered by an organization. See <i>Application and Business Services (EAM)</i> on page 35.
System			A packaged application. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Application			A computer program. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Database			A database. See <i>Systems, Applications, and Databases (EAM)</i> on page 38.
Component			A replaceable part of an application. See <i>Components (EAM)</i> on page 41.
Contract			An agreement between services. See <i>Contracts (EAM)</i> on page 48.
Application Link			A link between elements in an application layer diagram. See <i>Application Links (EAM)</i> on page 46.

Note: Most EAM objects can be displayed as icons or as boxes that can display additional properties and sub-objects. You can toggle between the Icon and Detail modes via the contextual menu, or by pressing **Ctrl+Q**. You can control the default mode, and the information displayed in each mode through the options available by selecting **Tools > Display Preferences**.

Application and Business Services (EAM)

An application service is an externally visible unit of functionality, provided by one or more applications or components, and exposed through well-defined interfaces. A business service is a service offered by an organization to its customers that directly supports the work performed in a business process or function, exposed by an application-to-business interface.

Business and application services can be created in the following diagrams:

- [application service only] Application architecture diagram (see *Application Architecture Diagrams* on page 29)
- Service-oriented diagram (see *Service-Oriented Diagrams* on page 32)
- City planning diagram (see *City Planning Diagrams* on page 14)

Example

In this example the Register sales business service is implemented by the Finance Service application service:



Creating an Application or Business Service

You can create an application or business service from the Toolbox, Browser, or **Model** menu.

- Use the **Application** or **Business Service** tool in the Toolbox.
- Select **Model > Application Services** or **Business Services** to access the appropriate list of services, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Application Service** or **Business Service**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Application and Business Service Properties

To view or edit an application or business service's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	<p>Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.</p> <p>Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.</p>
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Type	<p>Specifies the type of service.</p> <p>For business services, you can choose between:</p> <ul style="list-style-type: none"> • Consulting • Customer Service • Distribution Service • Marketing Service • Personal Service • Sales Service <p>For application services, you can choose between:</p> <ul style="list-style-type: none"> • Business Process • Collaboration • Data • Infrastructure • Integration • Presentation

Property	Description
Quality of Service Level	Specifies the quality of the service. You can choose between: <ul style="list-style-type: none"> • Best effort service • Differentiated service • Guaranteed service
Security Level	Specifies the security of the service. You can choose between: <ul style="list-style-type: none"> • High • Moderate • Low
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Role associations – lists the organization units, people, and roles associated with the service, and the type of role they play in relation to it (see *Role Associations* on page 20).
- Operations – lists the operations (see *Operations* on page 37) that support the service.

Creating Business and Application Service Operations

An operation is an abstract description of an action supported by a service. For example the Login service may require a Get ID operation. You can create operations from the property sheet of, or in the Browser under, a business or application service.

- Right-click a business or application service in the Browser, and select **New > Operation**.
- Click the Add a Row tool on the Operations tab of a business or application service property sheet.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Operation Properties

To view or edit an operation's properties, double-click its Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Parent	Specifies the service to which the operation belongs.

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Operation type	Specifies the type of the operation. You can choose between: <ul style="list-style-type: none"> • Notification • One-way • Request-Response • Solicit-Response
Input	Specifies the document or data required to start the service, which can be linked to an XML model to define its format (see <i>Forms, Documents, and Reports (EAM)</i> on page 42)
Output	Specifies the document or data returned by the service, which can be linked to an XML model to define its format (see <i>Forms, Documents, and Reports (EAM)</i> on page 42)
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

Systems, Applications, and Databases (EAM)

A system is a packaged application that can contain sub systems, applications, services and other application elements. An application is a computer program, a collection of data groups (databases and/or parts of databases) and program modules (components, database management systems, utilities, interactive modules, and system modules), which implements a business function. A database is a representation of an installed database.

These objects can be created in the following diagrams:

- City planning diagram (see *City Planning Diagrams* on page 14)
- Application architecture diagram (see *Application Architecture Diagrams* on page 29)
- Service-oriented diagram (see *Service-Oriented Diagrams* on page 32)

Example

In this example, the Inventory system contains the Stock Management application and the Stock database:



Creating a System, Application, or Database

You can create a system, application or database from the Toolbox, Browser, or **Model** menu.

- Use the appropriate tool in the Toolbox.
- Select **Model > Systems, Applications, or Databases** to access the appropriate object list, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > System, Application or Database**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

System, Application, and Database Properties

To view or edit a system, application, or database's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.

Property	Description
Type	<p>[application and database only] Specifies the type of application or database. For applications, you can choose between:</p> <ul style="list-style-type: none"> • Web • Rich Client • Office • Tool • CRM • ERP <p>For databases, you can choose between:</p> <ul style="list-style-type: none"> • Data Warehouse • Data Mart • Multi-Dimensional Data Warehouse • OLTP Database • Virtual Database
DBMS	[database only] Specifies the DBMS of the database.
DBMS version	[database only] Specifies the version of the DBMS.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Detail – [application only] records the programming language, version, etc, for the application.
- Role Associations – lists the people, organization units, and roles associated with the application or system, and the type of role they play in relation to it (see *Role Associations* on page 20).
- Source Models – [database only] lists the data models associated with the database. For more information about data models, see *Data Modeling*.
- Applications – [application and system only] lists the applications associated with the application or system.
- Components – [application and system only] lists the components (see *Components (EAM)* on page 41) associated with the application or system.
- Forms – [application and system only] lists the forms (see *Forms, Documents, and Reports (EAM)* on page 42) associated with the application or system.
- Sites – lists the sites associated with the application or system (see *Sites (EAM)* on page 22).
- Application Services – [system only] lists the application services (see *Application and Business Services (EAM)* on page 35) associated with the system.

- Databases – [system only] lists the databases associated with the system.
- Documents – [system only] lists the documents (see *Forms, Documents, and Reports (EAM)* on page 42) associated with the system.
- ETL Jobs – [system only] lists the ETL jobs (see *ETL Jobs (EAM)* on page 45) associated with the system.
- Report Documents – [system only] lists the reports (see *Forms, Documents, and Reports (EAM)* on page 42) associated with the system.
- Systems – [system only] lists the sub-systems contained within the system.

Components (EAM)

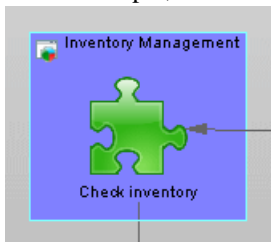
A component is an encapsulated, reusable, and replaceable part of an application, which can be used to implement a service or an application.

A component can be created in the following diagrams:

- Application architecture diagram (see *Application Architecture Diagrams* on page 29)
- Service-oriented diagram (see *Service-Oriented Diagrams* on page 32)

Example

In this example, the Inventory Management system contains the Check Inventory component:



Creating a Component

You can create a component from the Toolbox, Browser, or **Model** menu.

- Use the **Component** tool in the Toolbox.
- Select **Model > Components** to access the List of Components, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Component**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Component Properties

To view or edit a component's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Type	Specifies the type of the component. You can choose between: <ul style="list-style-type: none"> • EJB • Servlet • Presentation • DataAccess • Controller • COM • ActiveX
Programming Language	Specifies the programming language in which the component is written.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

Forms, Documents, and Reports (EAM)

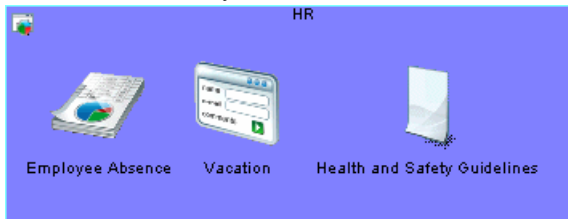
A form represents a UI component of an application. A document represents any conceptual document used in the enterprise. A report represents any report used in the enterprise.

These objects can be created in the following diagrams:

- Application architecture diagram (see *Application Architecture Diagrams* on page 29)

Example

In this example the HR system contains the Employee Absence report, the Vacation form, and the Health and Safety Guidelines document:

**Creating a Form, Document, or Report**

You can create a form, document or report from the Toolbox, Browser, or **Model** menu.

- Use the appropriate tool in the Toolbox.
- Select **Model > Forms, Documents, or Reports** to access the appropriate object list, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Form, Document or Report**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Form, Document, and Report Properties

To view or edit a form, document, or report's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Type	Specifies the type of the document, form, or report.
Version	[document only] Specifies the version number of the document.

Property	Description
Category	[report only] Specifies the category of the report. You can choose between: <ul style="list-style-type: none"> • Business • Specification • Strategic • Technical
Period	[report only] Specifies the frequency with which the report is generated. You can choose between: <ul style="list-style-type: none"> • Daily • Weekly • Monthly • Yearly
Style	[form only] Specifies the style of the form.
Document Format	[document only] Specifies the XML model that is used to represent the structure of the document. Select an XSM from the list or use the tools to the right of the field to create a new XSM or view the properties of the currently selected one. For more information about XML models, see <i>XML Modeling</i> .
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Data – [report and document only] lists the data (see *Data* on page 44) associated with the document.

Creating Document Data

Data objects are pieces of information that can be assigned to reports and documents. You can create a data object by using the Add a Row tool on the Data tab of a report or document.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Data Properties

To view or edit a data's properties, double-click its Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

ETL Jobs (EAM)

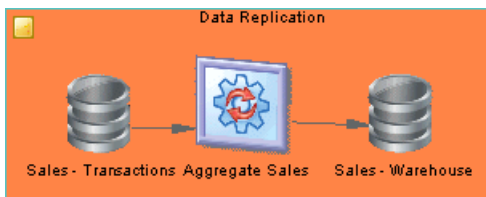
An ETL job represents a task in an ETL data transformation.

An ETL job can be created in the following diagrams:

- Application architecture diagram (see *Application Architecture Diagrams* on page 29)

Example

In this example the Aggregate Sales ETL job aggregates the data from the Sales – Transactions database to the Sales – Warehouse database:



Creating an ETL Job

You can create an ETL job from the Toolbox, Browser, or **Model** menu.

- Use the **ETL Job** tool in the Toolbox.
- Select **Model > ETL Jobs** to access the List of ETL Jobs, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > ETL Job**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

ETL Job Properties

To view or edit an ETL job's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Role Associations – lists the people, organization units, and roles associated with the ETL job, and the type of role they play in relation to it (see *Role Associations* on page 20).

Application Links (EAM)

An *application link* is an oriented link that is used to connect objects in the application layer.

An application link can be created in the following diagrams:

- Application architecture diagram (see *Application Architecture Diagrams* on page 29)
- Service-oriented diagram (see *Service-Oriented Diagrams* on page 32)

Example

In this example, the Create invoice component makes a request to the Invoice database:



Creating an Application Link

You can create an application link from the Toolbox, Browser, or **Model** menu.

- Use the **Application Link** tool in the Toolbox.
- Select **Model > Application Links** to access the List of Application Links, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Application Link**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Application Link Properties

To view or edit an application link's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Type	Specifies the type of the application link.

Property	Description
First object	Specifies the object at the origin of the link.
Second object	Specifies the object at the end of the link.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

Contracts (EAM)

A contract is an agreement between services.

A contract can be created in the following diagrams:

- Service-oriented diagram (see *Service-Oriented Diagrams* on page 32)

Example

In this example, the contract defines acceptable response times for the link between the business and application login services:



Creating a Contract

You can create a contract from the Toolbox, Browser, or **Model** menu.

- Use the **Contract** tool in the Toolbox.
- Select **Model > Contracts** to access the List of Contracts, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Contract**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Contract Properties

To view or edit a contract's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Role Associations – lists the people, organization units, and roles associated with the contract, and the type of role they play in relation to it (see *Role Associations* on page 20).
- Documents – lists the documents (see *Forms, Documents, and Reports (EAM)* on page 42) associated with the contract.
- XML Models – lists the XML Models that are used to define the contract. For more information about XSMs, see *XML Modeling*.

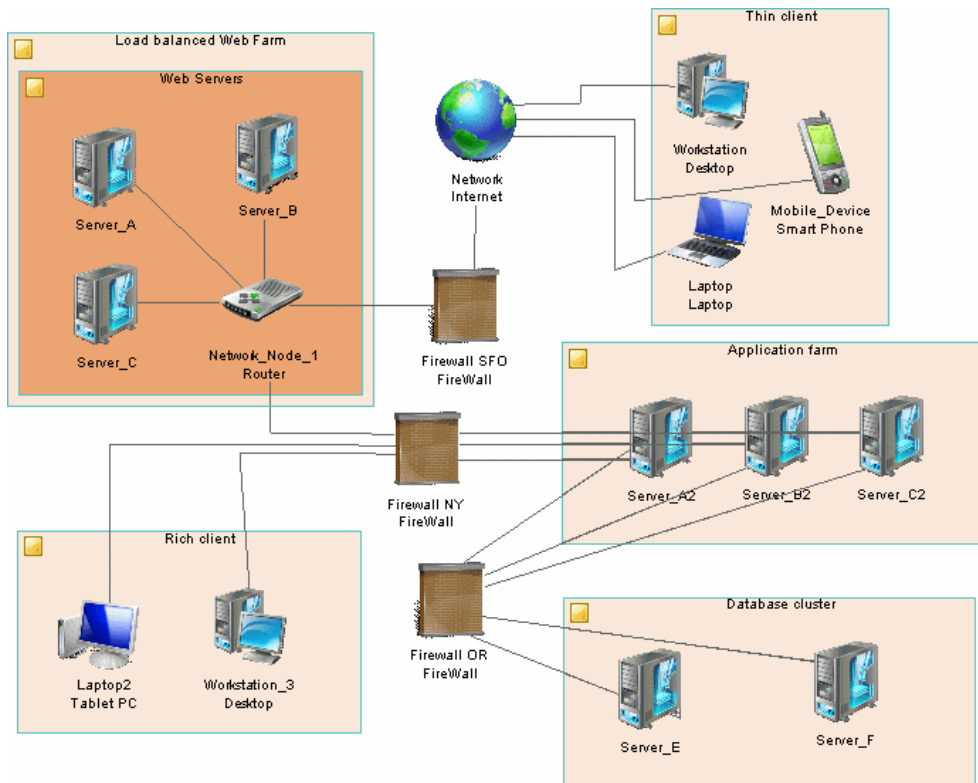
CHAPTER 4 Technology Layer Diagrams

A *technology infrastructure diagram* provides a high-level graphical view of the physical architecture required to support the application architecture.

Note: To create a technology infrastructure diagram in an existing EAM, right-click the model in the Browser and select **New > Technology Infrastructure Diagram**. To create a new model, select **File > New Model**, choose Enterprise Architecture Model as the model type and **Technology Infrastructure Diagram** as the first diagram, and then click **OK**.





















A technology infrastructure diagram is the sole diagram in the technology layer.







In the following example, the deployment of servers within the company network is shown, along with the web access afforded to thin clients:



Technology Infrastructure Diagram Objects

PowerDesigner supports all the objects necessary to build technology infrastructure diagrams.

Object	Tool	Symbol	Description
Architecture Area			An abstract object for grouping other objects. See <i>Architecture Areas (EAM)</i> on page 20.
Site			A physical location. See <i>Sites (EAM)</i> on page 22.
Organization Unit			A group, department, function, or other collection of people or organization units. See <i>Organization Units (EAM)</i> on page 15.
Person			An individual. See <i>People (EAM)</i> on page 17.
Role			A set of responsibilities. See <i>Roles (EAM)</i> on page 18.
Network			Any kind of network. See <i>Networks (EAM)</i> on page 53.
Software Server			A commercial software environment. See <i>Servers, Workstations, Mobile Devices, and Network Nodes (EAM)</i> on page 54.
Hardware Server			A high availability resource machine. See <i>Servers, Workstations, Mobile Devices, and Network Nodes (EAM)</i> on page 54.
Workstation			A client machine. See <i>Servers, Workstations, Mobile Devices, and Network Nodes (EAM)</i> on page 54.
Mobile Device			A handheld client. See <i>Servers, Workstations, Mobile Devices, and Network Nodes (EAM)</i> on page 54.

Object	Tool	Symbol	Description
Network Node			Can represent various kinds of network components. See <i>Servers, Workstations, Mobile Devices, and Network Nodes (EAM)</i> on page 54.
Deployment Instance			An instance of an object defined elsewhere. See <i>Deployment Instances (EAM)</i> on page 57.
Infrastructure Link			A link between elements in a technology layer diagram. See <i>Infrastructure Links (EAM)</i> on page 59.

Note: Most EAM objects can be displayed as icons or as boxes that can display additional properties and sub-objects. You can toggle between the Icon and Detail modes via the contextual menu, or by pressing **Ctrl+Q**. You can control the default mode, and the information displayed in each mode through the options available by selecting **Tools > Display Preferences**.

Networks (EAM)

A network allows computers to communicate with each other either locally or over large distances via telecommunications.

A network can be created in the following diagrams:

- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, the client communicates with the server via an intranet:



Creating a Network

You can create a network from the Toolbox, Browser, or **Model** menu.

- Use the **Network** tool in the Toolbox.
- Select **Model > Networks** to access the List of Networks, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Network**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Network Properties

To view or edit a network's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Type	Specifies the type of the network.
Site	Specifies the site within which the network is located.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- **Networks** - lists the sub-networks contained in the network.
- **Role Associations** – lists the people, organization units, and roles associated with the network (see *Role Associations* on page 20).
- **Hardware Servers, Software Servers, and Workstations** - list the servers and workstations associated with the network (see *Servers, Workstations, Mobile Devices, and Network Nodes (EAM)* on page 54).

Servers, Workstations, Mobile Devices, and Network Nodes (EAM)

A software server is a commercial software environment, which contains a set of applications and which can be deployed to a hardware server. A hardware server is a device that serves data or other support to network devices and which usually has higher specifications than client devices. A workstation is a client machine to which an application or a server can be deployed. A mobile device is a portable client used remotely to access the network. Network nodes

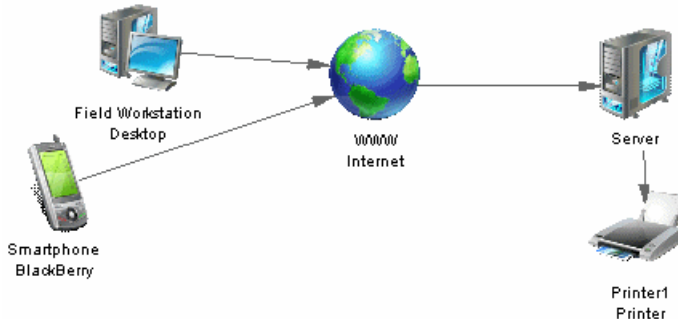
represent other type of network objects (routers, switches, firewalls, modems, printers, faxes, etc).

These objects can be created in the following diagrams:

- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, the field workstation and smartphone connect via the web to the server, which is also connected to a network node of type printer:



Creating a Software or Hardware Server, Workstation, Mobile Device, or Network Node

You can create a server, workstation, mobile device or network node from the Toolbox, Browser, or **Model** menu.

- Use the appropriate tool in the Toolbox.
- Select **Model > Objects** to access the appropriate object list, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Server, Workstation, Mobile Device or Network Node**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Software and Hardware Server, Workstation, Mobile Device, and Network Node Properties

To view or edit a software or hardware server, workstation, mobile device, or network node's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Type	Specifies the type of the object.
Site	Specifies the site to which the object is deployed.
Multiple	[hardware server, workstation and mobile device only] Specifies that the object represents multiple machines.
Virtual	[hardware server and workstation only] Specifies that the object represents a virtual machine. This property will be selected automatically if you create or drag the object onto a hardware or software server or workstation. For more information about creating virtual machines, see <i>Modeling Clusters and Virtual Machines</i> on page 56.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

- Detail – contains information to identify the object in the environment.
- Deployment Instances – [not network node] lists the deployment instances (see *Deployment Instances (EAM)* on page 57) associated with the object.
- Software Servers – [workstation, hardware server, and mobile device only] lists the software servers associated with the object.
- Role Associations – lists the people, organization units, and roles associated with the object (see *Role Associations* on page 20).

Modeling Cluster Servers and Virtual Machines

You can model complex machine-in-machine environments using the **Type** and **Virtual** properties.

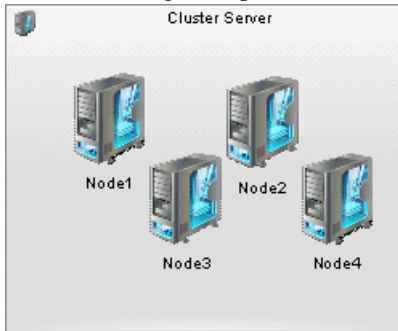
To model virtual machines, create or drag a hardware server or workstation onto a hardware or software server or a workstation. PowerDesigner will select and render read-only the **Virtual** property for objects modeled in this way. You can also model virtual machines in isolation by simply selecting the **Virtual** property yourself.

In the following example the hardware server, workstation, and software server each contain a virtual server and virtual workstation:



To model a cluster server, create a hardware server and set its **Type** to `Cluster Server`. You can then drop other hardware servers onto the symbol (or create them on the cluster server's **Hardware Servers** tab) to create cluster nodes. Nodes created in this way are not treated as virtual by default.

In the following example, the cluster server contains four hardware server nodes:



Deployment Instances (EAM)

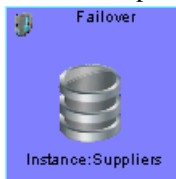
A deployment instance is an instance of another object that is deployed on a server or elsewhere in the environment.

A deployment instance can be created in the following diagrams:

- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, an instance of the Suppliers database is stored on the Failover server:



Creating a Deployment Instance

You can create a deployment instance from the Toolbox, Browser, or **Model** menu.

- Open the property sheet of the object to deploy and use the **Add a Row** tool on its **Deployment Instances** tab.
- Use the **Deployment Instance** tool in the Toolbox.
- Select **Model > Deployment Instances** to access the List of Deployment Instances, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Deployment Instance**.

Note: In all but the first method above, you must specify the object of which you are creating an instance, by opening the deployment instance property sheet and specifying the object of which it is an instance in the Deployed object field.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Deployment Instance Properties

To view or edit a deployment instance's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Deployed Object	Specifies the object being deployed. Click the buttons to the right of the field to create a new object, select an existing object, or to open the property sheet of the selected object.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

Infrastructure Links (EAM)

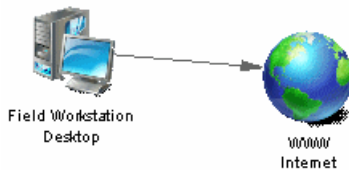
An *infrastructure link* is an oriented link that is used to connect objects in the technology layer.

An infrastructure link can be created in the following diagrams:

- Technology infrastructure diagram (see *Chapter 4, Technology Layer Diagrams* on page 51)

Example

In this example, the Field Workstation accesses the internet:



Creating an Infrastructure Link

You can create an infrastructure link from the Toolbox, Browser, or **Model** menu.

- Use the **Infrastructure Link** tool in the Toolbox.
- Select **Model > Infrastructure Links** to access the List of Infrastructure Links, and click the **Add a Row** tool.
- Right-click the model (or a package) in the Browser, and select **New > Infrastructure Link**.

For general information about creating objects, see *Core Features Guide > Modeling with PowerDesigner > Objects*.

Infrastructure Link Properties

To view or edit an infrastructure link's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Stereotype	Extends the semantics of the object. You can enter a stereotype directly in this field, or add stereotypes to the list by specifying them in an extension file.
Type	Specifies the type of the infrastructure link.
First object	Specifies the object at the origin of the link.
Second object	Specifies the object at the end of the link.
Protocol	Specifies the protocol of the infrastructure link.
Physical type	Specifies the physical type of the infrastructure link.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

Sybase IQ Reference Architecture Model

PowerDesigner provides a special EAM model to help you analyze the architecture required to deploy a Sybase® IQ data warehouse solution suitable for your workload. An advisor wizard generates architectures based on one or more hardware servers, and comparison tools help you choose the best architecture based on your requirements for cost and speed.

Creating a Reference Architecture Model

PowerDesigner provides a model template from which to create a reference architecture model, which by default is available in the **Technology** category of the **New Model** dialog.

1. Select **File > New Model**, and click the **Categories** button.
2. Select the **Technology** category and then select the **Sybase IQ Reference Architecture** template.

Note: If your **New Model** dialog is configured in such a way that this category or template are not available, choose **Model Types > Enterprise Architecture Model > Technology Infrastructure Diagram** and then click the **Select Extensions** button and attach the Sybase IQ Reference Architecture extension.

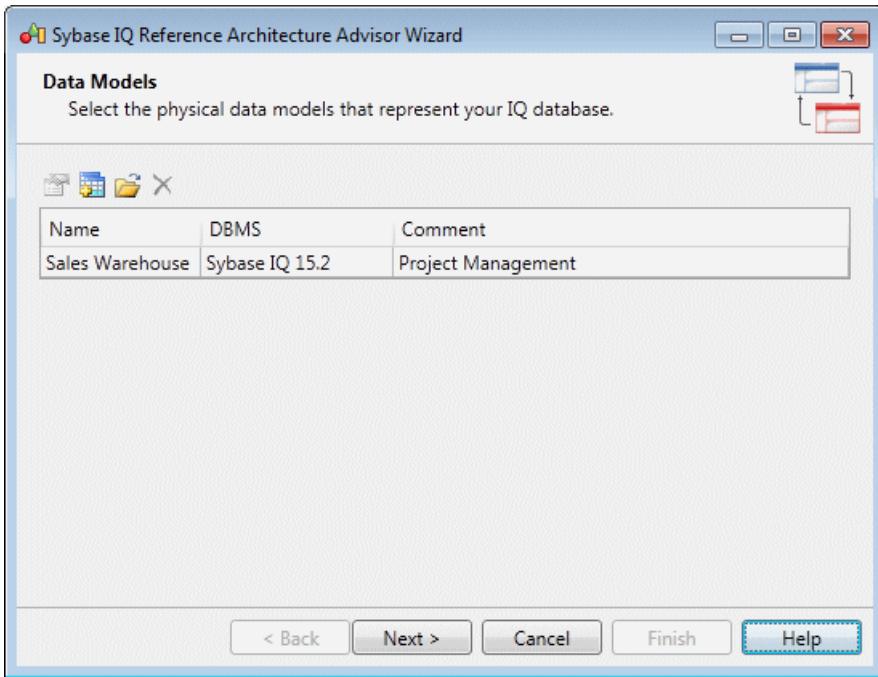
3. Enter a name for your model and click **OK** to create and open it.

Creating Reference Architectures

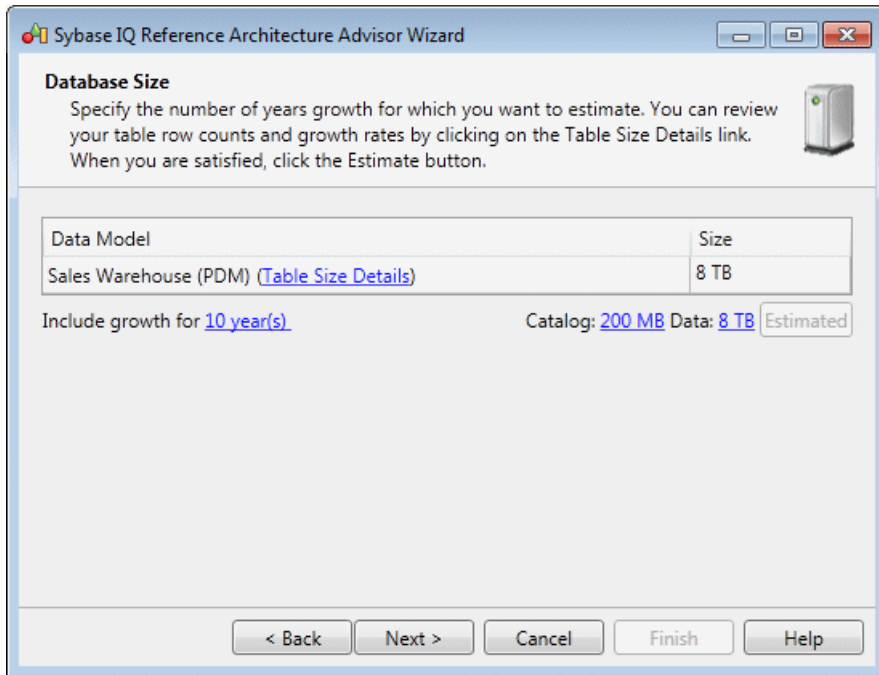
PowerDesigner provides a wizard to let you generate one or more reference architectures appropriate for your IQ environment, each based on a different hardware profile. The resulting diagram and objects detail the costs and specifications for all the elements required for each reference architecture.

Before launching the advisor wizard, you must develop a physical data model to represent your IQ database and have specified appropriate candidate machines in the hardware profile library (see *Chapter 6, Hardware Profile Library* on page 77).

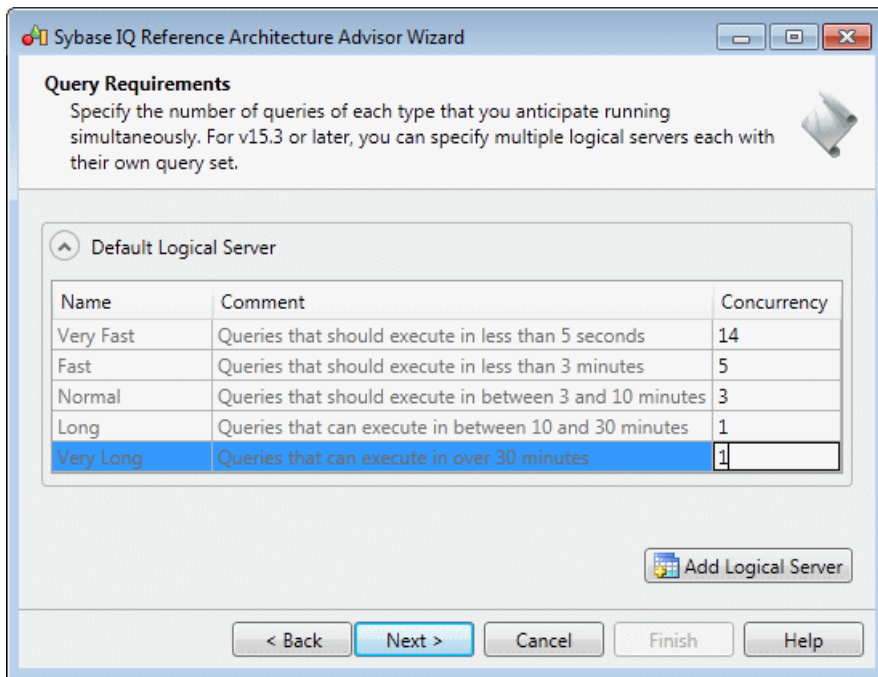
1. Select **Tools > Sybase IQ Reference Architecture > Launch Advisor Wizard** to open the wizard.
2. Click **Next** to go to the **Data Models** page, and use the **Select Data Models** tool to add a PDM from the workspace or the **Select Data Models from Files** tool to open a PDM:



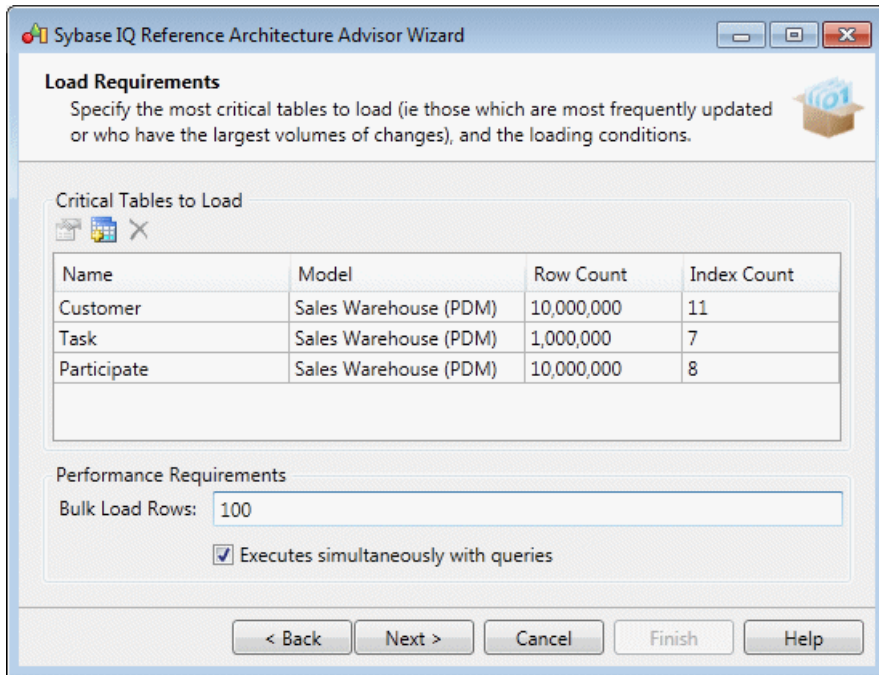
3. Click **Next** to go to the **Database Size** page, and specify the number of years growth for which you want to estimate. To review and edit table row counts and growth rates, click the **Table Size Details** link. When you are satisfied with your figures, click the **Estimate** button to produce an estimate of the overall size of your database at the end of the specified number of years:



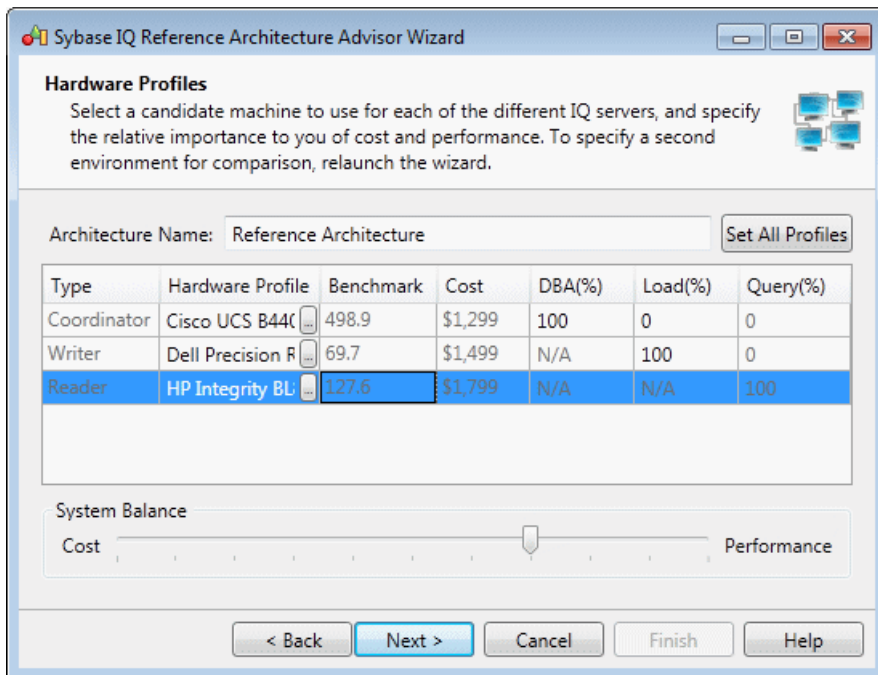
4. Click **Next** to go to the **Query Requirements** page, and specify the number of queries of each type that you anticipate running simultaneously:



5. [optional, v15.3 and higher] Click **Add Logical Server** to define another set of query requirements. You can add as many logical servers as necessary to accurately represent your environment.
6. Click **Next** to go to the **Load Requirements** page. Click the **Select Tables** tool to specify the most critical tables to load (those which have the largest volumes of changes). Then specify the number or rows that will be loaded in batches, and indicate whether this loading will take place while queries are being run:



7. Click **Next** to go to the **Hardware Profiles** page. Click the ellipsis tool to select a candidate machine from the Hardware Profile Library for the coordinator, writer, and reader. You can reallocate part of the DBA, load, and query work between these machines as necessary. Use the **System Balance** slider to specify the relative importance to you of cost and performance:

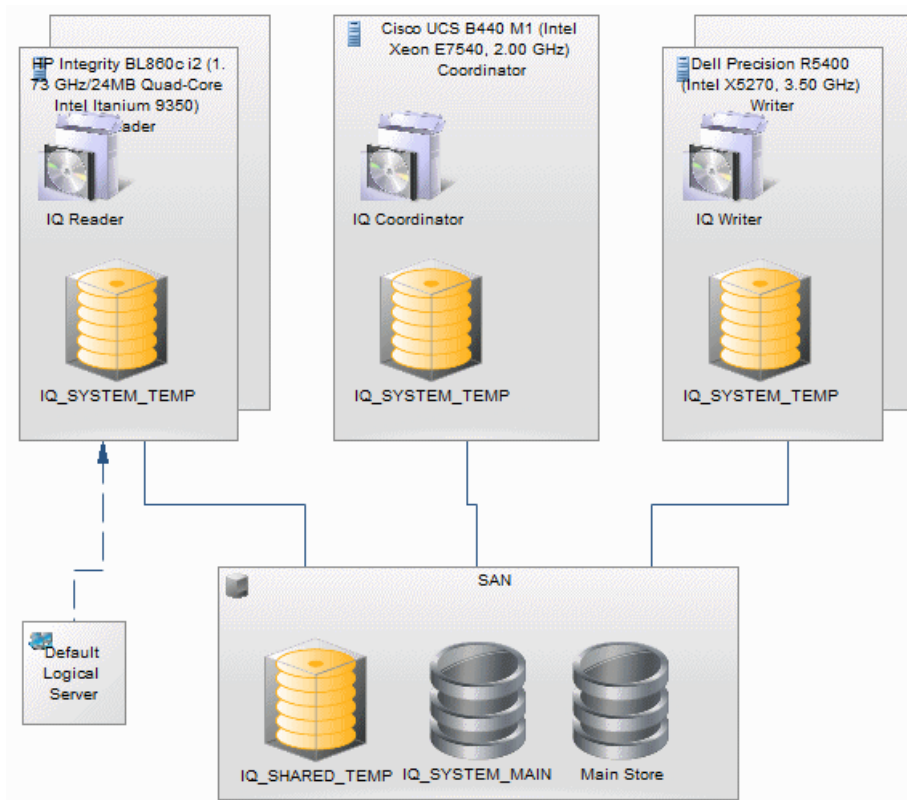


8. Click **Next** to go to the **Summary** page, which lists the information entered on the preceding pages.
9. Click **Finish** to generate your environments.

The Summary page lists the characteristics of the reference architecture, giving the number of nodes and the total cost of the solution. Click **Back** if you want to change any of your parameters in the wizard.

10. Click **Close** to exit the wizard and go to the generated diagram:

Double-click the reference architecture symbol to go to a sub-diagram showing the architecture in detail, with separate symbols for the SAN, coordinator, readers and writers:



Note: To model additional environments for comparison, relaunch the wizard and specify different hardware profiles.

Reference Architecture Properties

To view or edit a reference architecture's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

Note: If you modify any of these properties (which are generated by the Advisor wizard), you should regenerate the architecture to verify that your environment remains valid (see *Modifying and Regenerating Reference Architectures* on page 74).

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Summary	[read-only] Provides an overview of the costs, nodes, and storage sizes of the architecture.

The following tabs are also available:

- **Database Size** – Lists the size of the database, and the period to which the estimate applies. If you change the number of maintenance years, click the **Estimate** button to recalculate the database size.
- **Hardware Profile** – Lists the hardware profiles acting as coordinator, writer and reader, and provides a breakdown of their workloads. The system balance can range between 0% (low cost most important) and 100% (high performance most important).
- **Load Requirements** – Lists the most frequently updated tables and those that have the largest volume of changes, and specifies the IQ loading requirements.
- **Data Models** – Lists the PDMs that represent your IQ database.
- **Query Requirements** – Lists the number of each type of query that must be supported concurrently.

Database Server Properties

To view or edit a database server's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.

Property	Description
Hardware profile	[read-only] Specifies the hardware profile (see <i>Hardware Profile Properties</i> on page 78), which defines the server, its cost, the number of instances, and the total cost.

The following tabs are also available:

- **Detail** – [read-only] Provides information about the CPU, cores, memory, storage, file system, and operating system for the server.
- **IQ Servers** – Lists the IQ servers installed on the server (see *IQ Server Properties* on page 69).
- **IQ Stores** – Lists the IQ stores defined on the server (see *IQ Store Properties* on page 71).

IQ Server Properties

To view or edit an IQ server's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Type	Specifies whether the IQ server is a coordinator, reader, or writer.
Cost	Specifies the cost of the IQ server.

The **Detail** tab contains the following properties:

Property	Description
IQ Page Size	Specifies the page size in KB that is used for the IQ main store and temporary store disk devices.
IQ Catalog Page Size	Specifies the catalog page size in KB for moving data from memory to disk.

Property	Description
Catalog Cache	Specifies the size in MB of memory allocated for catalog operations. The catalog memory will also control, to a degree, how many connections and users can run queries simultaneously in IQ.
Bitmap Memory	Specifies the size in MB of bitmap memory, which is allocated for storing bitmap information.
Main Cache	Specifies the size in MB of the main cache, which is used to hold static, persistent user data and the system structures controlling access to the user data in the form of the IQ indexes.
Temp Cache	Specifies the size in MB of the temporary cache, which is used to hold temporary tables, internal work tables, and any other data structures that are temporary in nature.

SAN Properties

To view or edit a SAN's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Cost	[read-only] Specifies the total cost of the SAN, based on the price specified on the model property sheet (see <i>Reference Architecture Model Properties</i> on page 72).

The following tabs are also available:

- **Detail** – Lists the sizes of the IQ stores, application store, and backup store, along with the RAID level and access bandwidth in MB/s.
- **IQ Stores** – Lists the IQ stores defined on the SAN (see *IQ Store Properties* on page 71).

IQ Store Properties

To view or edit an IQ store's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Store type	Specifies whether the store is a catalog, main, or temp store.
Dbfile size (MB)	Specifies the size of each dbfile in MBs, the number of dbfiles, and the total dbfile size in GBs.

Query Type Properties

To view or edit a query type's properties, double-click its Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Parent object	Specifies the reference architecture for which the query type is defined.
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Concurrency	Specifies the number of this type of query that the environment must be able to perform simultaneously.

Reference Architecture Model Properties

A reference architecture model has more properties than a standard EAM.

The **General** tab contains all the standard model properties (see *EAM Properties* on page 5) in addition to the following properties:

Property	Description
Hardware profile library	Specifies the path to the hardware profile library. If no path is specified, the library will be searched for in the %_LIBRARY% named path.

Configuration Tab

These properties are available on the **Configuration** tab:

Property	Description
Main cache ratio for reader/ writer (%)	Specifies the percentage of the cache allocated to the main (as opposed to temporary) cache.
Memory margin (%)	Specifies the percentage of memory reserved for the operating system, file system, and other applications.
Application storage (GB)	Specifies the number of gigabytes of storage to be allocated for other applications.
Backup storage ratio (%)	Specifies the percentage of storage allocated to backup data. This number can be greater than 100%.
IQ temp store per user (GB)	Specifies the disk size (in GB) allocated for each concurrent user for the temp store.
SAN disk type	Specifies the type of disk used in the SAN.
SAN disk per core	Specifies the standard and maximum number of disk devices in the SAN system per CPU core.
IQ_SYS-TEM_MAIN (GB)	Specifies the standard and maximum sizes (in GB) allocated to the IQ_SYS-TEM_MAIN dbspace.
Memory per core (GB)	Specifies the standard and maximum memory (in GB) per core.
Main cache per index for writer (IQ page)	Specifies the standard and maximum main cache (in IQ pages) per index for writers.

Logical Server Properties

To view or edit a logical server's properties, double-click its Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Parent object	Specifies the reference architecture for which the query type is defined.
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Concurrency	Specifies the number of this type of query that the environment must be able to perform simultaneously.

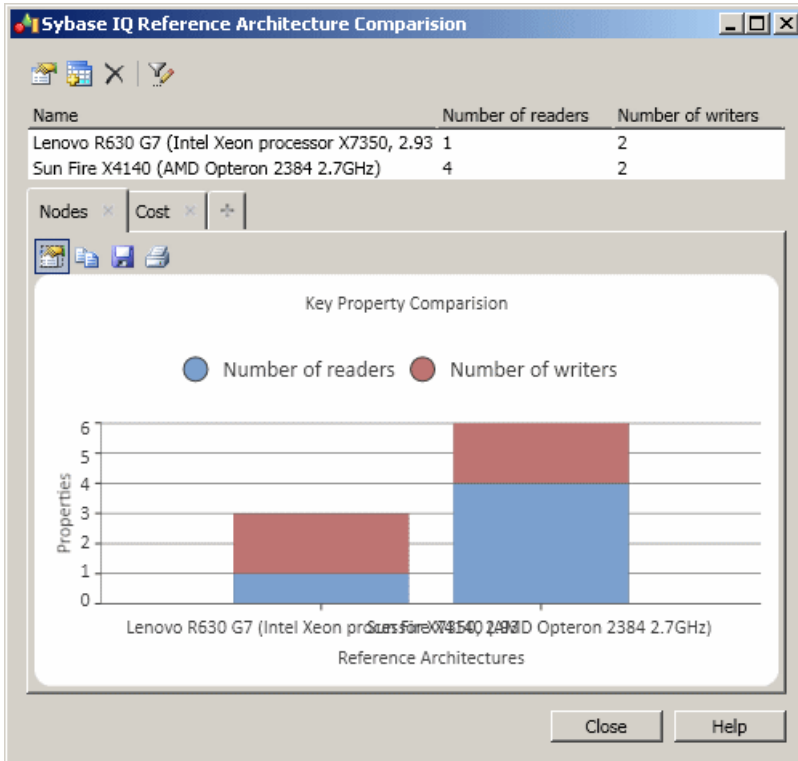
The following tabs are also available:

- **Query Requirements** - Lists the number of each type of query that must be supported concurrently.

Comparing Reference Architectures

PowerDesigner provides a wizard to let you compare any properties of multiple reference architectures using a range of chart styles.

1. Select **Tools > Sybase IQ Reference Architecture > Compare Reference Architectures** to open the wizard.
A default chart using the last chosen parameters appears.
2. [optional] Use the tools above the list to add or remove reference architectures for comparison, and to select properties to display in the list.
3. [optional] Click the **Chart Properties** tool above the chart to select the type of chart and select attributes to compare.
4. [optional] Click the plus sign next to the Chart tab to create addition charts:



5. [optional] Use the other chart tools to copy the chart to the clipboard, save the chart as an image, or print the chart.

Modifying and Regenerating Reference Architectures

You can modify the properties generated by the Advisor wizard (for example to specify different hardware profiles for reader and writer nodes), but after having done so, you should regenerate the architecture to recalculate the other parameters and verify that your architecture is still valid.

1. Modify any of the generated properties on the reference architecture property sheet (see *Reference Architecture Properties* on page 67), and then click **OK** to save your changes.

For example, you could change the number of years for which you want to estimate growth, change the hardware profile for the coordinator, writer, or reader, or modify your query or loading requirements.

Note: If you change the number of years for which you want to estimate growth, you must click the Estimate button to recalculate the database size before regenerating.

2. Right-click the reference architecture in the Browser or diagram and select **Sybase IQ Reference Architecture > Regenerate**

A model check is performed to ensure that the architecture is still valid and, if this is the case, it is regenerated.

Generating Reports from a Reference Architecture Model

You can generate reports from your reference architecture model by selecting **Tools > Sybase IQ Reference Architecture > Generate HTML Report** or **Generate RTF Report**.

PowerDesigner provides a hardware profile library (in the `Library` folder in the installation directory) to store detailed technical information about hardware servers that can be used to model IQ reference architectures or as shortcuts in other models.

Importing Profiles from the SPEC Website

PowerDesigner provides sample hardware profiles downloaded from the Standard Performance Evaluation Corporation (SPEC) website (www.spec.org), with SPECint2006 Rates benchmark data. You can download additional profiles from the site and import them into your library.

Note: You can also create a hardware profile from the List of Hardware Profiles (**Tools > Hardware Profiles**) and entering the properties by hand, but we recommend importing the profiles from the SPEC site.

1. In your Web browser, go to the benchmark search form <http://www.spec.org/cgi-bin/osgresults?conf=rint2006&op=form>
2. Set the following columns to `Display`:
 - Hardware Vendor
 - System
 - # Cores
 - # Chips
 - # Cores Per Chip
 - # Threads Per Core
 - Processor
 - Processor MHz
 - Memory
 - Operating System
 - File System
 - Published
3. Enter any appropriate criteria. For example, you may want to restrict the search results to servers from your approved vendors, or to those available since the beginning of the year.
4. Click **Fetch Results** to perform the search.
The results are returned on a new page.
5. To download all the results, click the **Download** link, next to the Results heading. To download an individual server result, click the **CSV** link to the right of its entry. Both links provide CSV files that can be imported into your hardware profile library.

Note: We advise you to download individual profiles, as they provide more detailed information, including storage size.

6. Right-click the Hardware Profile Library model in the Browser and select **Hardware Profiles > Import**
7. Click the **Select CSV Files** tool and navigate to and select the CSV files you downloaded from the SPEC Website.
8. Click **Import** to import the profiles into the library and then **Close** to close the wizard.
9. Open the property sheets of the imported servers to add the appropriate unit cost to your organization.

Note: If you do not specify costs for your hardware profiles, you will not obtain useful costing information from the IQ reference architecture Advisor wizard (see *Creating Reference Architectures* on page 61).

Hardware Profile Properties

To view or edit a hardware profile's properties, double-click its diagram symbol or Browser or list entry. The property sheet tabs and fields listed here are those available by default, before any customization of the interface by you or an administrator.

The **General** tab contains the following properties:

Property	Description
Name/Code/ Comment	Identify the object. The name should clearly convey the object's purpose to non-technical users, while the code, which is used for generating code or scripts, may be abbreviated, and should not normally include spaces. You can optionally add a comment to provide more detailed information about the object. By default the code is generated from the name by applying the naming conventions specified in the model options. To decouple name-code synchronization, click to release the = button to the right of the Code field.
Vendor	Specifies the vendor of the server.
Cost	Specifies the cost of the server. This value is used to calculate the overall cost of the environment.
Benchmark/ Type	Specifies the server's benchmark result and the type of benchmark.

The following tabs are also available:

- **Detail** – Provides information about the CPU, cores, memory, storage, file system, and operating system for the server.

Hardware Profile Library Properties

A hardware profile library has more properties than a standard EAM.

The **General** tab contains all the standard model properties (see *EAM Properties* on page 5) in addition to the following properties:

Property	Description
Currency	Specifies the currency to use in cost calculations.
SAN cost per TB	Specifies the cost of a terabyte of SAN storage.

Sybase IQ Configuration Tab

These properties are available on the **Sybase IQ Configuration** tab:

Property	Description
Reference Profile	Specifies the hardware profile that is used as a baseline against which to measure the cost and performance of other profiles.
Load cores per...	Specifies the standard and maximum number of cores required for each column, HG, WD, and other index when loading.
Bandwidth per core (MB/s)	Specifies the standard and maximum disk bandwidth (in MB/s) per core.
Reference Query Baselines	Specifies the supported number of concurrent queries for each query type when the system balance is set to 0% (low cost most important) and 100% (high performance most important).

Specifying Additional Query Types

You can specify additional query types that must be supported by your hardware profiles.

1. Open the property sheet of the hardware profile library and click the **Sybase IQ Configuration** tab.
2. In the **Query Type Baselines** list, click the **Add a Row** tool, and enter a name and comment to identify and describe your new query type.
3. In the **Minimum Concurrency** column, enter the number of simultaneous instances of the query that the reference profile can support (in addition to all the other query types) when the system balance is set to 0% (low cost most important).
4. In the **Maximum Concurrency** column, enter the number of simultaneous instances of the query that the reference profile can support (in addition to all the other query types) when the system balance is set to 100% (high performance most important).

CHAPTER 6: Hardware Profile Library

The next time that you launch the Advisor wizard in an reference architecture model associated with the hardware library (see *Creating Reference Architectures* on page 61), you will be able to specify requirements for the new query type.

The enterprise architecture model is a very flexible tool, which allows you quickly to develop your model without constraints. You can check the validity of your EAM at any time.

A valid EAM conforms to the following kinds of rules:

- Each object name must be unique in a EAM
- Each network in must be linked to another object
- Each person must belong to an organization unit
- Each software server must be deployed to a physical server

Note: We recommend that you check your enterprise architecture model before generating another model from it . If the check encounters errors, generation will be stopped. The **Check model** option is enabled by default in the Generation dialog box.

You can check your model in any of the following ways:

- Press F4, or
- Select **Tools > Check Model**, or
- Right-click the diagram background and select Check Model from the contextual menu

The Check Model Parameters dialog opens, allowing you to specify the kinds of checks to perform, and the objects to apply them to. The following sections document the EAM -specific checks available by default. For information about checks made on generic objects available in all model types and for detailed information about using the Check Model Parameters dialog, see *Core Features Guide > Modeling with PowerDesigner > Objects > Checking Models*.

Package Checks

PowerDesigner provides default model checks to verify the validity of packages.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> • Manual correction - Modify the name or code to contain only glossary terms. • Automatic correction - None.

Check	Description and Correction
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
List of missing terms in object names/codes in package	<p>[if glossary enabled] This special check lists all the terms that are used in the names or codes of all the objects contained in the model or package.</p> <ul style="list-style-type: none"> Manual correction - An administrator with write permission on the glossary can double-click the line for this check in the Result List to open a merge window in which you can select to add some or all of these missing terms, as appropriate, to the glossary. Automatic correction - None.

Application, Application Service, Component and Database Checks

PowerDesigner provides default model checks to verify the validity of applications, application services, components, and databases.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.

Check	Description and Correction
Name/Code uniqueness	Object names must be unique in the namespace. <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
No deployment instance	The object must be deployed as at least one deployment instance. <ul style="list-style-type: none"> Manual correction: Deploy the object as a deployment instance. Automatic correction: None

Application Link Checks

PowerDesigner provides default model checks to verify the validity of application links.

Check	Description and Correction
Name/Code contains terms not in glossary	[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary. <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.
Name/Code contains synonyms of glossary terms	[if glossary enabled] Names and codes must not contain synonyms of glossary terms. <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	Object names must be unique in the namespace. <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
Unsuitable application link type	The link type is not compatible with the link extremities. For example, if the link type is "use" the link must have a role player as one extremity. <ul style="list-style-type: none"> Manual correction: Change the link type. Automatic correction: Changes the link type.

Contract Checks

PowerDesigner provides default model checks to verify the validity of contracts.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
Contracts are not correctly used	<p>Contracts must have at least one incoming and one outgoing link with an application service, a business service, or a role.</p> <ul style="list-style-type: none"> Manual correction: Create input and output links with the appropriate objects. Automatic correction: None
Multiple input service types	<p>Contracts must have either roles, business services, or application services as inputs, but not any combination of these types.</p> <ul style="list-style-type: none"> Manual correction: Remove any input links that are not of the same type. Automatic correction: None
Same input and output link	<p>Contracts cannot be the intermediary between the same service.</p> <ul style="list-style-type: none"> Manual correction: Modify the input or output link. Automatic correction: None

Deployment Instance Checks

PowerDesigner provides default model checks to verify the validity of deployment instances.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
Deployment Instance not deployed	<p>Each deployment instance must be deployed to a platform or server.</p> <ul style="list-style-type: none"> Manual correction: Deploy the deployment instance to the appropriate object. Automatic correction: None
No deployed object specified	<p>The deployment instance must be an instance of some object.</p> <ul style="list-style-type: none"> Manual correction: Specify the object to be deployed in the deployment instance property sheet General tab. Automatic correction: None
Unsuitable Software Server	<p>The deployment instance must be suitable for the server software type to which it is deployed.</p> <ul style="list-style-type: none"> Manual correction: Modify the software server type or delete the deployment instance from the software server. Automatic correction: None

Form Checks

PowerDesigner provides default model checks to verify the validity of forms.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
No deployment instance	<p>The object must be deployed as a deployment instance.</p> <ul style="list-style-type: none"> Manual correction: Deploy the object as a deployment instance. Automatic correction: None

Hardware Server, Mobile Device, Network Node and Workstation Checks

PowerDesigner provides default model checks to verify the validity of hardware servers, mobile devices, network nodes and workstations.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.

Check	Description and Correction
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
Serial number uniqueness	<p>Serial numbers must be unique in the model.</p> <ul style="list-style-type: none"> Manual correction: Modify the duplicate serial numbers. Automatic correction: Appends a number to the duplicate serial numbers.
Asset number uniqueness	<p>Asset numbers must be unique in the model.</p> <ul style="list-style-type: none"> Manual correction: Modify the duplicate asset numbers. Automatic correction: Appends a number to the duplicate asset numbers.
Mac address uniqueness	<p>Mac addresses must be unique in the model.</p> <ul style="list-style-type: none"> Manual correction: Modify the duplicate mac addresses. Automatic correction: Appends a number to the duplicate mac addresses.
Isolated node	<p>The object must be linked with at least one other node.</p> <ul style="list-style-type: none"> Manual correction: Link the object to an appropriate object. Automatic correction: None

Network Checks

PowerDesigner provides default model checks to verify the validity of networks.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.

Check	Description and Correction
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
Isolated node	<p>The object must be linked with at least one other node.</p> <ul style="list-style-type: none"> Manual correction: Link the object to an appropriate object. Automatic correction: None

Organization Unit Checks

PowerDesigner provides default model checks to verify the validity of organization units.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
Manager not belonging to the managed organization	<p>The manager specified in the Manager field must be listed in the People tab.</p> <ul style="list-style-type: none"> Manual correction: Add the manager in the list of people on the People tab. Automatic correction: Adds the manager to the list of people.

Person Checks

PowerDesigner provides default model checks to verify the validity of people.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - None.
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> Manual correction - Modify the name or code to contain only glossary terms. Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> Manual correction - Modify the duplicate name or code. Automatic correction - Appends a number to the duplicate name or code.
No organization unit	<p>A person must belong to an organization unit.</p> <ul style="list-style-type: none"> Manual correction: Add the person to an organization unit. Automatic correction: None
Circular dependency through manager property	<p>A person cannot manage herself.</p> <ul style="list-style-type: none"> Manual correction: Remove the person from the Manager field of her own property sheet. Automatic correction: None
Manager must belong to the same organization unit	<p>A person's manager must belong to the same organization unit as her.</p> <ul style="list-style-type: none"> Manual correction: Assign the manager to the same organization unit as the person being managed. Automatic correction: None

Software Server Checks

PowerDesigner provides default model checks to verify the validity of software servers.

Check	Description and Correction
Name/Code contains terms not in glossary	<p>[if glossary enabled] Names and codes must contain only approved terms drawn from the glossary.</p> <ul style="list-style-type: none"> • Manual correction - Modify the name or code to contain only glossary terms. • Automatic correction - None.
Name/Code contains synonyms of glossary terms	<p>[if glossary enabled] Names and codes must not contain synonyms of glossary terms.</p> <ul style="list-style-type: none"> • Manual correction - Modify the name or code to contain only glossary terms. • Automatic correction - Replaces synonyms with their associated glossary terms.
Name/Code uniqueness	<p>Object names must be unique in the namespace.</p> <ul style="list-style-type: none"> • Manual correction - Modify the duplicate name or code. • Automatic correction - Appends a number to the duplicate name or code.
Software Server not deployed	<p>A software server must be deployed to a physical server.</p> <ul style="list-style-type: none"> • Manual correction: Deploy the software server in the appropriate object. • Automatic correction: None
Software Server serial number uniqueness	<p>Software Server serial numbers must be unique in the model.</p> <ul style="list-style-type: none"> • Manual correction: Modify the duplicate serial numbers. • Automatic correction: Appends a number to the duplicate serial numbers.

Importing to and Generating and Exporting from an EAM

PowerDesigner provides various tools to import to and generate and export from your EAM.

Importing Visio Diagrams into PowerDesigner

Importing your Visio diagrams into PowerDesigner's rich metadata environment enables you to link your architectural objects with the objects that will implement them, and to profit from PowerDesigner's powerful impact and lineage analysis features. You must have installed Visio 2002 or higher and have selected to install the Visio plug-in from the PowerDesigner installer.

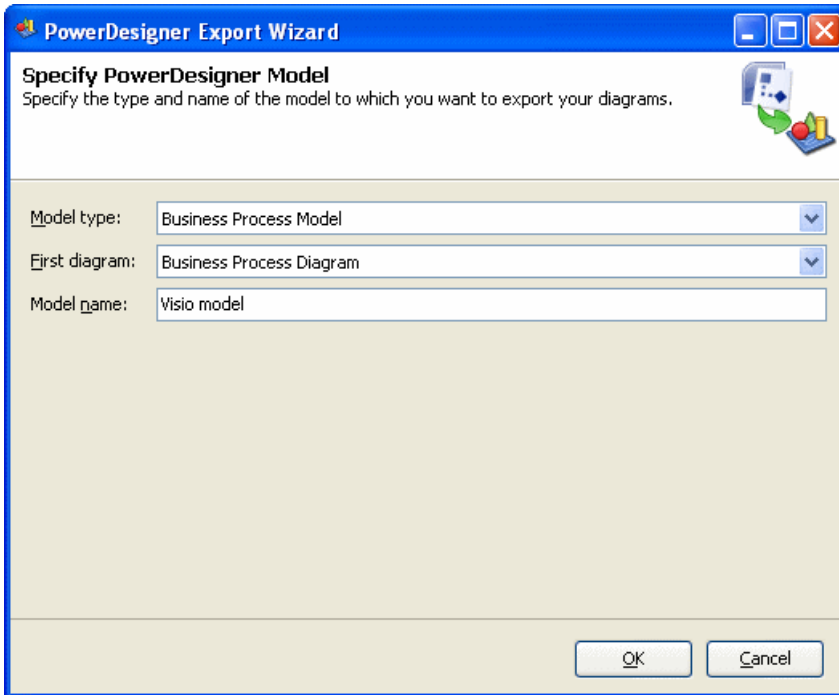
Note: Only Visio diagrams created from the following standard templates can be imported into PowerDesigner, and only objects available on the standard stencils will be imported. Custom properties will be imported as extended attributes.

You can import the following diagrams into a PowerDesigner BPM or EAM:

Visio Template	PowerDesigner Diagram
Audit Diagram	BPM Analysis/ Business Process Diagram
Basic Flowchart	BPM Analysis/ Business Process Diagram
Cross-Functional Flowchart	BPM Analysis/ Business Process Diagram
Business Process/ Data Flow Diagram Software/ Data Flow Diagram	BPM Data Flow Diagram
Event Driven Process Chain Diagram	BPM Business Process Diagram
ITIL Diagram	BPM Business Process Diagram
Work Flow Diagram	BPM Business Process Diagram
Flowchart/ SDL Diagram	BPM Business Process Diagram
Organization Chart	EAM Organization Chart Diagram
Software/ Enterprise Application	EAM Application Architecture Diagram
Network/ Basic Network / Detailed Network Diagram	EAM Technology Infrastructure Diagram
Active Directory	EAM Organization Chart Diagram

Visio Template	PowerDesigner Diagram
LDAP Directory	EAM Organization Chart Diagram

1. Open your diagram in Visio and select **PowerDesigner > Export to PowerDesigner Model** to open the PowerDesigner Export wizard:



2. Specify the type of model to which you want to export your diagram, enter a name for the model to be created, and then click **OK** to start the export.
3. When the export is complete, click **OK** to close the wizard.

The diagram is opened as a new BPM or EAM in PowerDesigner.

Exporting and Importing Objects to and from Other Models

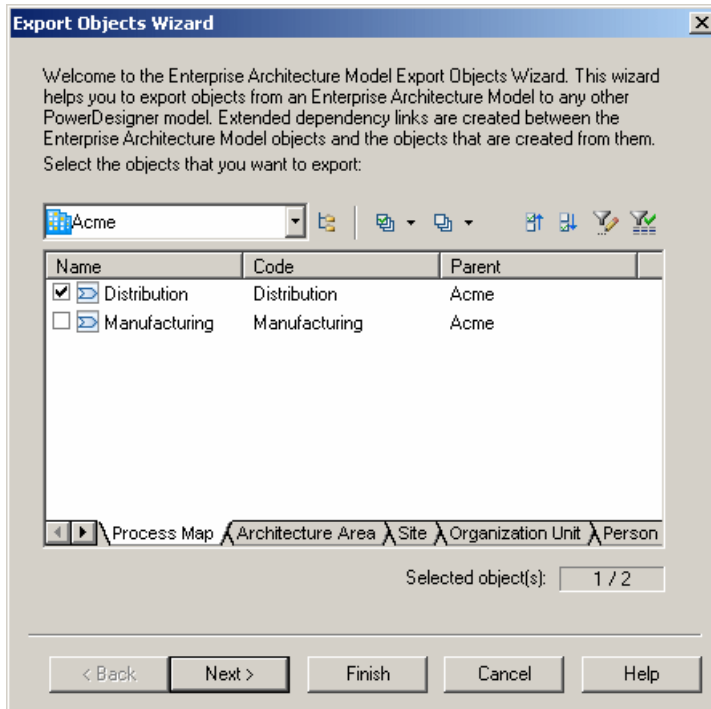
An enterprise architecture model is intended to provide a big picture view of your organization and to permit you to decompose your functions, processes, and systems to a certain extent, but it cannot show detailed information about how they are (or will be) implemented. In order to model such information, you will want to use other PowerDesigner models, such as the conceptual, logical, physical data models, business process models, or object-oriented models.

The EAM provides wizards to export EA objects to these other, lower-level models and to import implementation objects to be represented in the EAM.

Exporting Objects from an EAM to Another Model

The Export Objects Wizard helps you to export EAM objects to other PowerDesigner models to allow you to model their implementation in detail.

1. Select **Tools > Export Objects** to open the Export Objects Wizard:



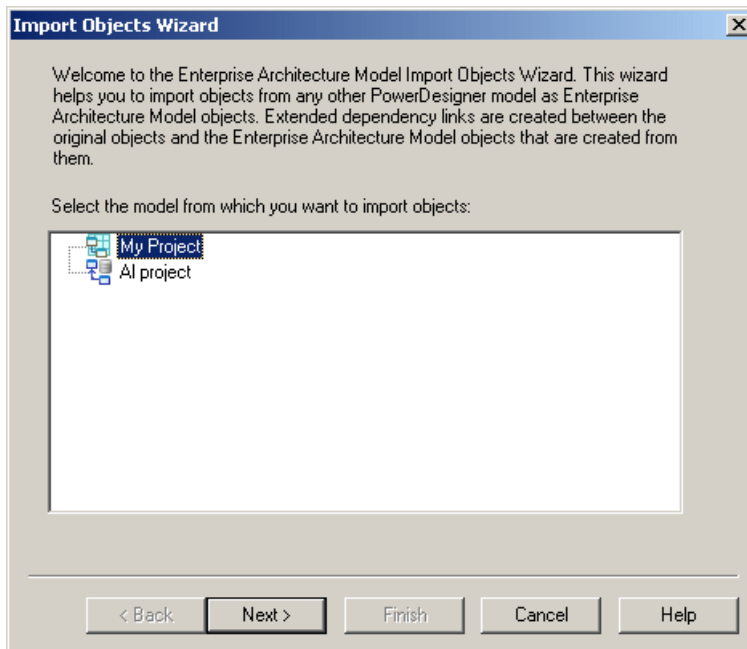
2. Specify the EAM objects that you want to export to another model by selecting them from among the sub-tabs in the selection field, and then click Next.
3. Specify the model to which you want to export the EAM objects from among the models open in the workspace, and then click Next.
4. Specify the kind of object that you want to create in the target model.
5. [optional] Enter or select a stereotype to identify the type of dependency that will be created between the EAM objects and the objects that you are creating from them. For example, "Implemented by".
6. Click Finish to start the export.

The EAM objects are exported to and created in the specified model.

Importing Objects from Another Model into an EAM

The Import Objects Wizard helps you to import objects from other PowerDesigner models into an EAM to model their placement in your enterprise architecture.

1. Select **Tools > Import Objects** to open the Import Objects Wizard:



2. Specify the model from which you want to import the objects to the EAM from among the models open in the workspace, and then click Next.
3. Specify the objects that you want to import to the EAM by selecting them from among the sub-tabs in the selection field, and then click Next.
4. Specify the kind of object that you want to create in the EAM and the location (package) where you want to create them.
5. [optional] Enter or select a stereotype to identify the type of dependency that will be created between the objects and the EAM objects that you are creating from them. For example, "Implements".
6. Click Finish to start the import.

The objects are imported to and created in the EAM.

Generating an EAM from an EAM

You can generate another EAM from your EAM. When changes are made to the source model, they can be easily propagated to any generated models by regenerating, using the Update Existing Model generation mode.

1. Select **Tools > Generate Enterprise Architecture Model** to open the EAM Generation Options Window:
2. On the General tab, select to generate a new or update an existing model, and complete the appropriate options.
3. [optional] Click the Detail tab and set any appropriate options. We recommend that you select the Check model checkbox to check the model for errors and warnings before generation.
4. [optional] Click the Target Models tab and specify the target models for any generated shortcuts.
5. [optional] Click the Selection tab and select or deselect objects to generate.
6. Click OK to begin generation.

Note: For detailed information about the options available on the various tabs of the Generation window, see *Core Features Guide > Linking and Synchronizing Models > Generating Models and Model Objects*.

Index

A

- application 38
 - check 82
 - create 39
 - properties 39
- application architecture diagram 29
 - objects 30
 - symbols 30
- application link 46
 - check 83
 - create 47
 - properties 47
- application service 35
 - check 82
 - create 35
 - properties 36
- architecture area 20
 - create 21
 - properties 21

B

- business communication diagram 10
 - objects 11
 - symbols 11
- business flow 26
 - create 27
 - properties 27
- business function 24
 - create 24
 - properties 24
- business service 35
 - create 35
 - properties 36

C

- check model 81
 - application 82
 - application link 83
 - application service 82
 - component 82
 - contract 84
 - database 82

- deployment instance 85
 - form 86
 - hardware server 86
 - mobile device 86
 - network 87
 - network node 86
 - organization unit 88
 - package 81
 - people 89
 - software server 90
 - workstation 86
- city planning diagram 14
 - objects 15
 - symbols 15
- cluster 56
- component 41
 - check 82
 - create 41
 - properties 42
- contract
 - check 84
 - create 48
 - properties 48

D

- data
 - create 44
 - properties 44
- database 38
 - check 82
 - create 39
 - properties 39
- database server
 - properties 68
- deployment instance 57
 - check 85
 - create 58
 - properties 58
- display preferences 7
 - symbols 10, 11, 13, 15, 30, 34, 52
- document 42
 - create 43
 - properties 43

Index

E

EAM

- application architecture 29
 - business communication diagram 10
 - check model 81
 - city planning diagram 14
 - create 3
 - organization chart 9
 - process map 12
 - service-oriented diagram 32
 - technology infrastructure diagram 51
- enterprise architecture framework project
- create 5
- ETL job 45
- create 45
- ETL Job
- properties 46
- export
- objects from other models 92
- extension 7
- extension file 7

F

- form 42
- check 86
 - create 43
 - properties 43

H

- hardware profile
- download 77
 - properties 78
- hardware profile library 77
- properties 79
 - query type 79
- hardware server 54
- check 86
 - cluster 56
 - create 55
 - properties 55

I

- import
- objects from other models 92
 - Visio 91

- infrastructure link 59
- create 59
 - properties 59

IQ server

- properties 69

IQ store

- properties 71

L

logical server

- properties 73

M

mobile device 54

- check 86
- create 55
- properties 55

model

- create 3
- model options 6
- properties 5
- model options 6
- modeling environment
- customize 6

N

network 53

- check 87
- create 53
- properties 54

network node 54

- check 86
- create 55
- properties 55

O

object

- application architecture diagram 30
- business communication diagram 11
- city planning diagram 15
- organization chart 10
- process map 13
- service-oriented diagram 34
- technology infrastructure diagram 52

- operation
 - create 37
 - properties 37
 - organization chart 9
 - objects 10
 - symbols 10
 - organization unit 15
 - check 88
 - create 16
 - properties 16
- P**
- package
 - check 81
 - people
 - check 89
 - person 17
 - create 17
 - properties 18
 - process 25
 - create 26
 - properties 26
 - process map 12
 - objects 13
 - symbols 13
- Q**
- query type
 - properties 71
- R**
- reference architecture
 - comparing 73
 - creating 61
 - generating report 75
 - modifying 74
 - properties 67
 - regenerating 74
 - reference architecture model
 - properties 72
 - report 42
 - create 43
 - properties 43
 - role 18
 - create 19
 - properties 19
 - role association 20
- S**
- SAN
 - properties 70
 - service-oriented diagram 32
 - objects 34
 - symbols 34
 - site 22
 - create 23
 - properties 23
 - software server 54
 - check 90
 - create 55
 - properties 55
 - Sybase IQ
 - reference architecture model 61, 73, 75, 77
 - symbol
 - application architecture diagram 30
 - business communication diagram 11
 - city planning diagram 15
 - detail mode 10, 11, 13, 15, 30, 34, 52
 - display preferences 10, 11, 13, 15, 30, 34, 52
 - icon mode 10, 11, 13, 15, 30, 34, 52
 - organization chart 10
 - process map 13
 - service-oriented diagram 34
 - sub-objects 10, 11, 13, 15, 30, 34, 52
 - technology infrastructure diagram 52
 - system 38
 - create 39
 - properties 39
- T**
- technology infrastructure diagram 51
 - objects 52
 - symbols 52
 - technology layer 61, 77
 - traceability link 8
- V**
- Visio
 - import 91
 - virtualization 56

Index

W

workstation 54
 check 86
 create 55
 properties 55

X

xem 7