

SAP® PowerDesigner®
Document Version: 16.5 SP05 – 2015-03-24

Enterprise Architecture Modeling



Content

1	Planning Your Enterprise Architecture Initiative.	5
1.1	Preparing Your PowerDesigner Landscape and Repository.	9
1.2	Creating an EAM.	10
	EAM Properties.	12
	Setting Model Options.	12
1.3	Creating an Enterprise Architecture Framework Project.	13
1.4	Creating EAM Diagrams.	15
	Displaying EAM Objects in Diagrams.	15
	Setting EAM Display Preferences.	19
1.5	Example EA Model, Imports, and Charts.	19
	Chart Examples.	21
1.6	Extending your Modeling Environment.	25
2	Organization Architecture Modeling.	27
2.1	Organization Charts.	27
2.2	Business Communications Diagrams.	28
2.3	Sites (EAM).	30
2.4	Organization Units (EAM).	32
2.5	People (EAM).	33
	Associating a Person, Organization Unit, or Role with an Object.	35
	Importing Role Associations from Excel.	37
2.6	Roles (EAM).	39
2.7	Business Flows (EAM).	40
2.8	Importing Organization Architecture Metadata from Excel.	41
3	Process and Function Architecture Modeling.	43
3.1	Process Maps.	43
	Creating a Multi-Level Process Map.	44
3.2	City Planning Diagrams.	45
3.3	Architecture Areas (EAM).	46
3.4	Business Functions (EAM).	48
3.5	Processes (EAM).	49
	Linking Processes to Business Process Diagrams.	51
3.6	Importing Process Architecture Metadata from Excel.	52
4	Application Architecture Modeling.	54
4.1	Application Architecture Diagrams.	54
4.2	Service-Oriented Diagrams.	56

4.3	Systems, Applications, and Databases (EAM)	58
4.4	Components (EAM)	61
4.5	Forms, Documents, and Reports (EAM)	62
	Document Data (EAM)	64
4.6	ETL Jobs (EAM)	65
4.7	Contracts (EAM)	66
4.8	Application and Business Services (EAM)	68
	Business and Application Service Operations	70
4.9	Application Links (EAM)	71
4.10	Importing Application Architecture Metadata from Excel	72
5	Infrastructure Architecture Modeling	74
5.1	Technology Infrastructure Diagrams	74
5.2	Servers, Workstations, Mobile Devices, and Network Nodes (EAM)	76
	Modeling Cluster Servers and Virtual Machines	78
5.3	Networks (EAM)	78
5.4	Deployment Instances (EAM)	80
	Importing Deployment Instances from Excel	82
5.5	Infrastructure Links (EAM)	83
5.6	Importing Infrastructure Architecture Metadata from Excel	84
6	Goal and EA Project Modeling	86
6.1	Timeline Diagrams	86
6.2	Goals (EAM)	89
6.3	Programs, Projects, and Phases (EAM)	91
	Specifying the Fulfillment of Goals by a Project	92
	Specifying the Impact of a Project on EA Assets	93
6.4	Importing Corporate Goals and EA Projects from Excel	94
	Importing Impacts, Fulfillments, and Stakeholders from Excel	95
7	SAP IQ Reference Architecture Model	98
7.1	Creating a Reference Architecture Model	98
7.2	Creating Reference Architectures	99
	Reference Architecture Properties	104
	Database Server Properties	105
	IQ Server Properties	106
	SAN Properties	107
	IQ Store Properties	107
	Query Type Properties	108
	Reference Architecture Model Properties	108
	Logical Server Properties	109
7.3	Comparing Reference Architectures	110

7.4	Modifying and Regenerating Reference Architectures	112
7.5	Generating Reports from a Reference Architecture Model	112
8	Hardware Profile Library	113
8.1	Importing Profiles from the SPEC Website	113
	Hardware Profile Properties	114
8.2	Hardware Profile Library Properties	115
	Specifying Additional Query Types	116
9	Checking an EAM	117
9.1	Package Checks	117
9.2	Application, Application Service, Component and Database Checks	118
9.3	Application Link Checks	119
9.4	Contract Checks	119
9.5	Deployment Instance Checks	120
9.6	Form Checks	121
9.7	Hardware Server, Mobile Device, Network Node and Workstation Checks	122
9.8	Network Checks	123
9.9	Organization Unit Checks	123
9.10	Person Checks	124
9.11	Software Server Checks	125
10	Importing and Linking to and Generating and Exporting from an EAM	126
10.1	Importing EA Metadata from Excel	126
10.2	Importing Visio Diagrams into PowerDesigner	127
10.3	Exporting and Importing Objects to and from Other Models	129
	Exporting Objects from an EAM to Another Model	130
	Importing Objects from Another Model into an EAM	132
10.4	Traceability Links	133
	Importing Traceability Links from Excel	134
10.5	Specifying Diagrams as Related Diagrams	135
10.6	Generating an EAM from an EAM	136

1 Planning Your Enterprise Architecture Initiative

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

Context

SAP® PowerDesigner® provides a robust solution for collecting, analyzing, communicating, and maintaining enterprise architecture metadata in order to:

- Align IT strategy with business goals, capabilities, and processes.
- Comply with legal standards or regulations requiring documentation of your organization's architecture.
- Document and communicate *change management* for:
 - Rationalization – seeking cost savings and efficiencies by standardizing and simplifying architectures.
 - Technology changes – where your system will be upgraded.
 - An acquisition or merger – where two disparate systems must be made to work together.

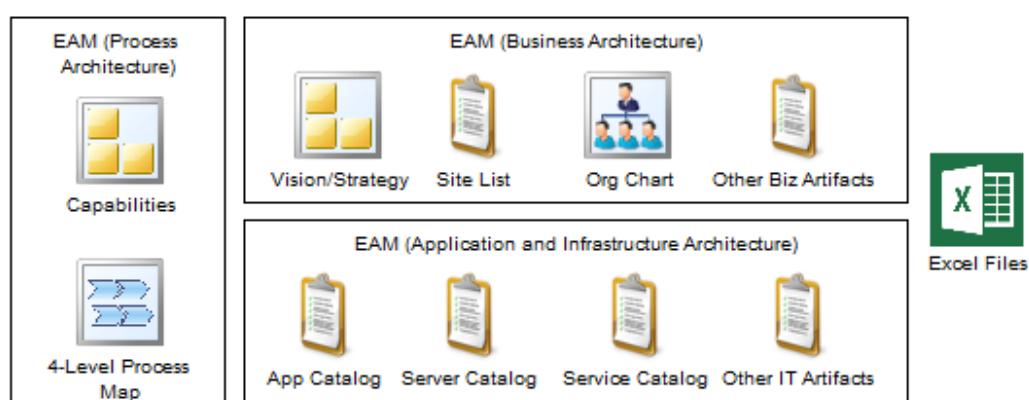
Procedure

1. Prepare your PowerDesigner landscape and repository to receive your metadata (see [Preparing Your PowerDesigner Landscape and Repository \[page 9\]](#)).
2. Decide what metadata you want to collect and how you want to structure it. The four broad domains of EA metadata are:
 - Organization Architecture - The people, sites, and structures of your organization (see [Organization Architecture Modeling \[page 27\]](#))
 - Process and Function Architecture - What your organization does and how it does it (see [Process and Function Architecture Modeling \[page 43\]](#))
 - Application Architecture - The software that your organization uses (see [Application Architecture Modeling \[page 54\]](#))
 - Infrastructure Architecture - The hardware and physical assets that your organization uses (see [Infrastructure Architecture Modeling \[page 74\]](#))

Note

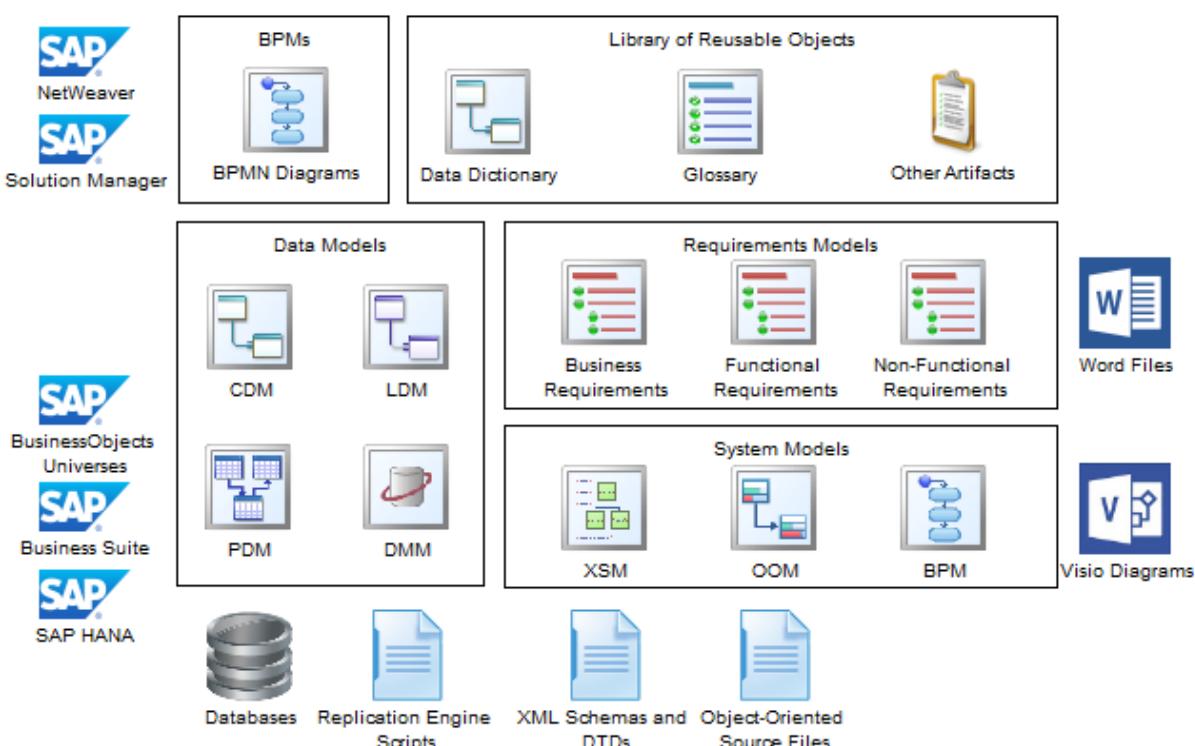
The EA_Example.eam uses a subset of the available EA objects from each of these domains, structured in a particular way, and can be used as the starting point for your deliberations (see [Example EA Model, Imports, and Charts \[page 19\]](#)).

3. Create individual models (see [Creating an EAM \[page 10\]](#)) or an enterprise architecture framework project that combines your EAM diagrams with those of a Physical Data Model and other PowerDesigner modules ([Creating an Enterprise Architecture Framework Project \[page 13\]](#)) to begin modeling.
4. Import previously collated metadata where possible (see [Importing EA Metadata from Excel \[page 126\]](#) and [Importing Visio Diagrams into PowerDesigner \[page 127\]](#)).

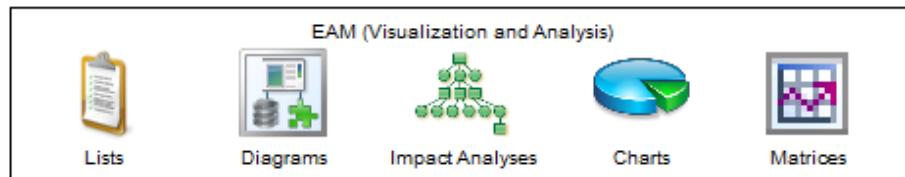


5. Connect your high-level EA models to your underlying landscape by inviting stakeholders, DBAs, developers, and implementers to create more detailed models of processes, databases, web services, and other assets.

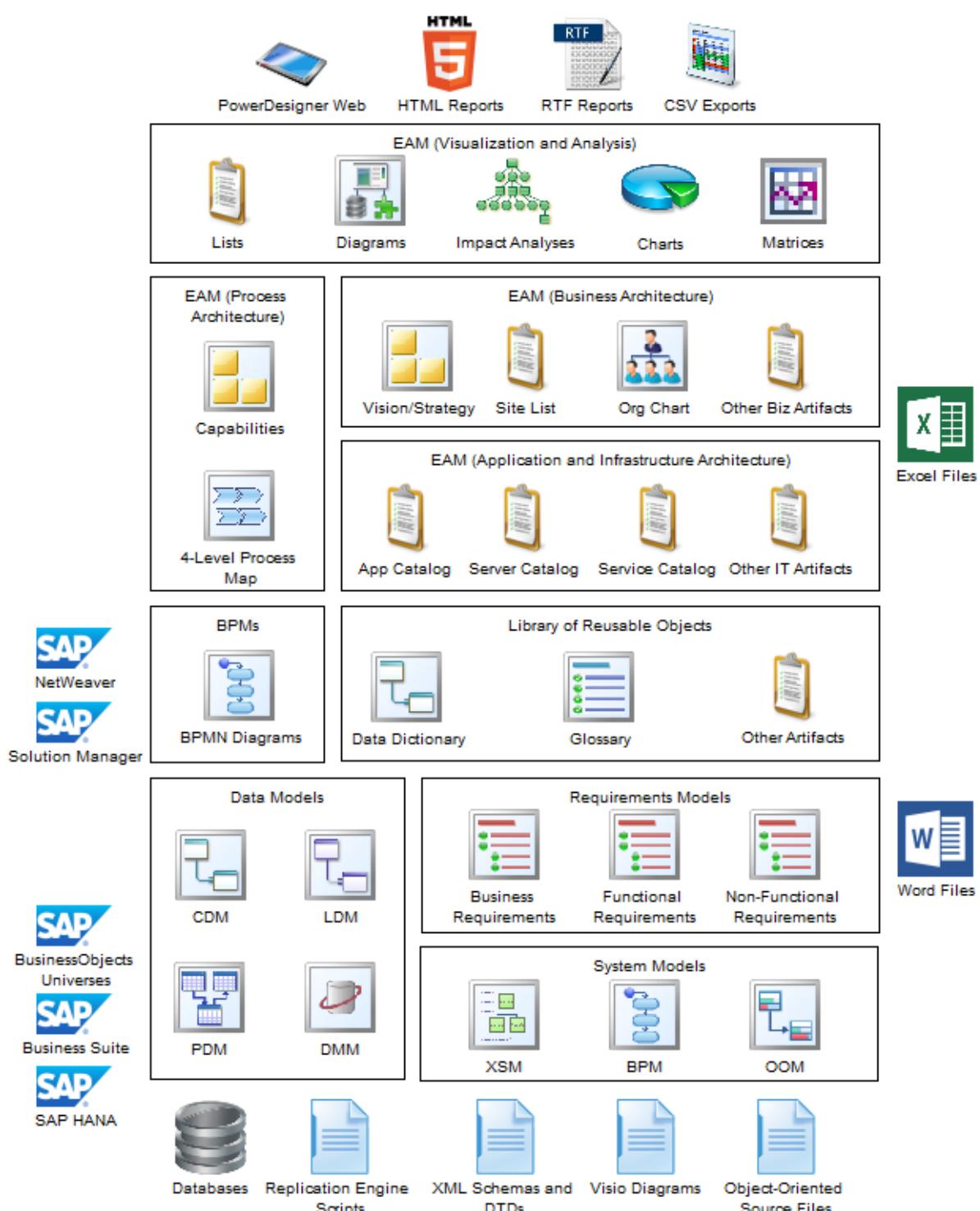
Business users who are process owners can document their processes in BPMN 2.0 business process diagrams using the Web browser-based PowerDesigner Web client (see *PowerDesigner Web*), while more technical users will use the PowerDesigner desktop client.



6. Create connections between your enterprise architecture and implementation models using:
 - Import and export wizards (see [Exporting and Importing Objects to and from Other Models \[page 129\]](#)).
 - Traceability links (see [Traceability Links \[page 133\]](#)).
 - Related diagrams (see [Specifying Diagrams as Related Diagrams \[page 135\]](#)).
7. Create appropriate viewpoints into your EA assets using:
 - Diagrams - Visualize and organize selected assets according to a particular need (see [Creating EAM Diagrams \[page 15\]](#) and [Displaying EAM Objects in Diagrams \[page 15\]](#)).
 - Charts - Analyze your assets based on datasets you define (see [Chart Examples \[page 21\]](#)).
 - Impact Analyses - Show how changes to one or more objects will impact others to which they are linked (see [Core Features Guide > Linking and Synchronizing Models > Impact and Lineage Analysis](#)).
 - Dependency Matrices - Show connections between different types of objects (see [Core Features Guide > Modeling with PowerDesigner > Diagrams, Matrices, and Symbols > Dependency Matrices](#)).



8. Add information about how your corporate goals can be fulfilled through changes to your landscape, and the impacts that such changes will have on your assets (see [Goal and EA Project Modeling \[page 86\]](#)).
9. Communicate your EA metadata to stakeholders and the wider organization, using:
 - PowerDesigner Web - Provides access to all your diagrams and metadata via a Web browser (see [PowerDesigner Web](#)).
 - HTML and RTF Reports - Provide snapshots of your metadata suitable for documentation or inclusion in other reports (see [Core Features Guide > Storing, Sharing and Reporting on Models > Reports](#)).
 - CSV Exports - For consumption by other reports or processes. Available from any list dialog.
10. Maintain your metadata in your repository and use it for ongoing planning, communication, and development:



1.1 Preparing Your PowerDesigner Landscape and Repository

The PowerDesigner repository provides a central point of storage for all your modeling assets, and can serve as the single source of truth for your enterprise metadata.

Procedure

1. Deploy your PowerDesigner repository to an appropriate server (see *Installation Guide > Planning Your PowerDesigner® Installation*) and connect to it.
2. Design an appropriate top-level folder structure for your repository to organize the models and other documents that will be stored there. For example, you may create the following folders:
 - Library - Pushes reference information and extensions and customization to all users (see *Core Features Guide > Modeling with PowerDesigner > The Browser > The Library*. Lists and catalogs of enterprise artifacts are commonly saved to the Library to promote their reuse in other models:
 - Glossary - Used for standardizing object names (see *Core Features Guide > Modeling with PowerDesigner > The Browser > The Glossary*).
 - Data Dictionary (CDM) - Defines common data elements (see *Data Modeling > Building Data Models > Conceptual and Logical Diagrams > Conceptual Diagrams > Example: Building a Data Dictionary in a CDM*).
 - Organization Architecture (EAM) - Sites, Organization Units, and People (see [Organization Architecture Modeling \[page 27\]](#)).
 - Resource Files - PowerDesigner resource files are often distributed through the library to ensure that all modelers are using the latest extensions and customizations (see [Extending your Modeling Environment \[page 25\]](#)).
 - EA Content - Can contain one or more high-level EA models and project documents, and may be branched to show the current or "as-is" situation and the projected "to-be" landscape.
 - EA Projects - Can contain one or more EA models drawing on objects in your landscape to show the goals that a project will contribute to fulfilling, and the assets that it will impact while doing so (see [Goal and EA Project Modeling \[page 86\]](#)).
 - Processes - Commonly contains an EAM model containing your 4-level process map and a folder structure to organize the business process models used to model in detail your level 4 processes (see [Process Maps \[page 43\]](#)).
 - Implementation Models - Can contain data models and other lower-level implementation models.
3. Create user accounts and set appropriate permissions based on who will need to consult, contribute to, or manage the metadata in each of your folders (see *Core Features Guide > Administering PowerDesigner > Controlling Repository Access*).

1.2 Creating an EAM

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

Context

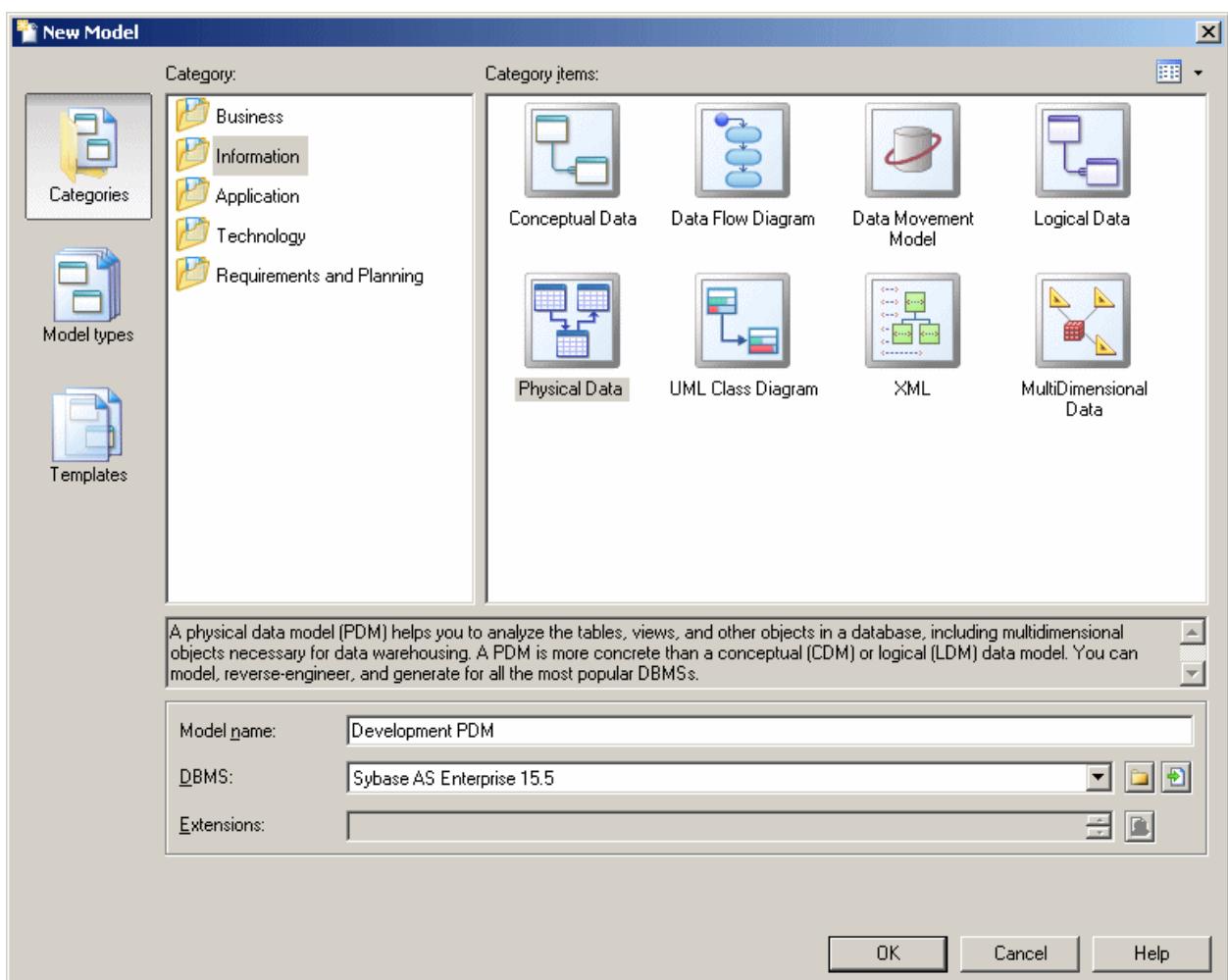
Note

In addition to creating an EAM from scratch with the following procedure, you can also create a model by importing:

- An Excel file (see [Importing EA Metadata from Excel \[page 126\]](#)).
- A Visio file (see [Importing Visio Diagrams into PowerDesigner \[page 127\]](#)).

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.



Procedure

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 .

i Note

Sample EAMs are available in the Example Directory.

1.2.1 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:

Table 1:

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 <i>Code</i> 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.

1.2.2 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:

Table 2:

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.

Option	Description
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 Shortcut 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, <i>All</i> properties appear, but you can select to display only <i>Name/Code</i> to reduce the size of your model.

i 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*.

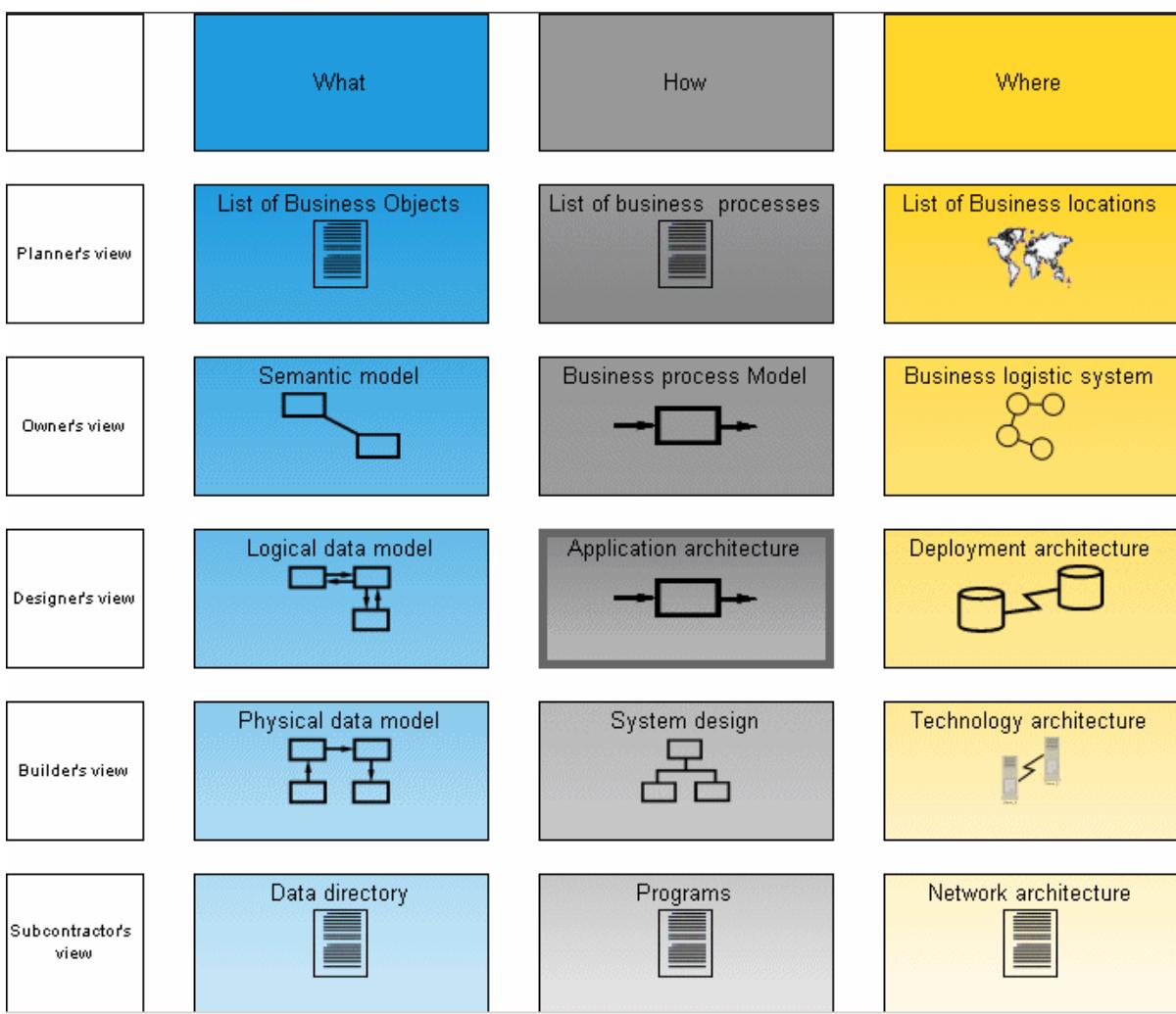
1.3 Creating an Enterprise Architecture Framework Project

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

Procedure

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:



Results

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

1.4 Creating EAM Diagrams

When you want to visualize a part of your environment for yourself or a stakeholder, create a diagram and add the appropriate objects to it. If another stakeholder requires a different level of detail or different information, then create a separate diagram for them.

i Note

Every PowerDesigner object can be displayed in multiple diagrams to show it in different contexts. All of the links that you create to other objects in all such diagrams are stored in the object to allow you to analyze the entirety of its dependencies (see *Core Features Guide > Linking and Synchronizing Models > Impact and Lineage Analysis*).

You specify a first diagram when you create an EAM. To add further diagrams in your model, right-click its Browser entry and select **New > Diagram**. The following types of diagrams are available:

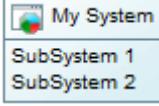
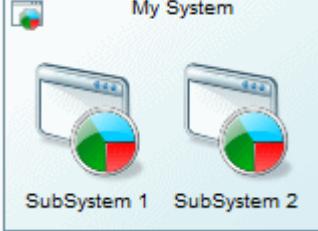
- A *process map* provides a graphical view of your business architecture, and helps you identify your business functions and high-level processes, independent of the people and business units who fulfill them. See [Process Maps \[page 43\]](#).
- A *city planning diagram* provides a graphical view of the big picture of your enterprise architecture, using the metaphor of planning the infrastructure of a city to represent the organization of functions, systems, applications, etc into architectural areas. See [City Planning Diagrams \[page 45\]](#).
- 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. See [Organization Charts \[page 27\]](#).
- 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. See [Business Communications Diagrams \[page 28\]](#).
- 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. See [Application Architecture Diagrams \[page 54\]](#).
- 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. See [Service-Oriented Diagrams \[page 56\]](#).
- A *technology infrastructure diagram* provides a high-level graphical view of the physical architecture required to support the application architecture. See [Technology Infrastructure Diagrams \[page 74\]](#).
- A *timeline diagram* provides a calendar-style overview of your enterprise architecture programs and projects in relation to your goals, or to the assets that they impact or are in other ways related to. See [Timeline Diagrams \[page 86\]](#).

1.4.1 Displaying EAM Objects in Diagrams

You can create EAM objects directly in a diagram using the Toolbox or *Model* menu. You can display existing objects in a diagram by dragging them from the Browser or selecting **Symbol > Show Symbol** and choosing them from the various sub-tabs of the *Show Symbols* dialog. For a list of the diagrams that can display a particular type of object, see the relevant object's topic.

The following symbol modes are available:

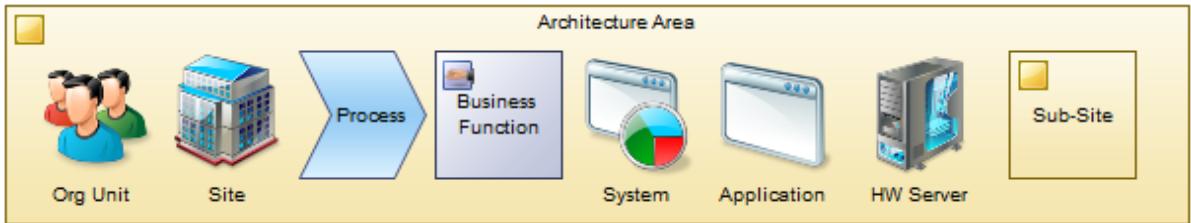
Table 3:

Mode	Details
Icon: 	Most EAM objects support icon or picture symbols that display a name or other limited information beneath them. To switch between icon and detail mode, right-click the symbol and toggle the <i>Show Detail</i> command or press CTRL+Q . To change any object symbol to an icon, select a <i>Custom Shape</i> of type Bitmap in the Symbol Format or Display Preferences dialog.
Detail: 	All objects support detail symbols (rectangles or other shapes) that can contain extensive information about the object inside the symbol, including lists of sub-objects. To control the information displayed, use the <i>Content</i> tab in the Symbol Format or Display Preferences dialog.
Editable Composite: 	Some EAM objects (see below) support editable composite symbols that allow you to create or drag and drop sub-objects on the symbol. To switch between composite and detail mode, right-click the symbol and use the <i>Composite View</i> submenu.

- To change the format or content of one or more individual symbols, select them, then right-click in the selection, and select *Format*. Use the commands *Get Format* and *Apply Format* to copy the format of one symbol to another.
- To change the format or content of all the symbols in a diagram, select ► *Tools* ► *Display Preferences* ▾.
- To arrange and manipulate groups of symbols, use the *Symbol* menu.

The following objects provide the principal ways for grouping other objects:

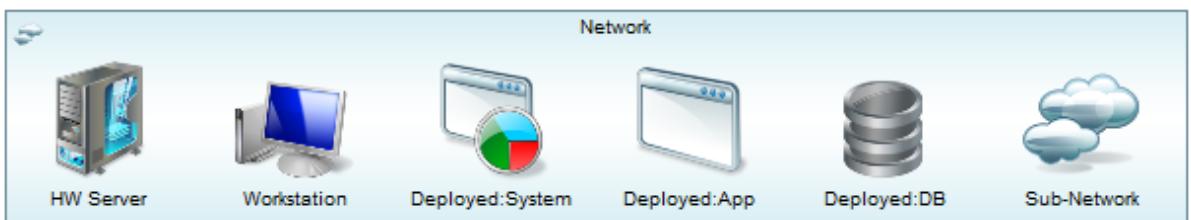
- Architecture areas - Provide logical groupings of objects (see [Architecture Areas \(EAM\) \[page 46\]](#)). Any object can be attached to an architecture area, can appear inside the area symbol and be listed on the area's property sheet *Attached Objects* tab. Objects can be attached to multiple architecture areas.



- Sites - Provide geographical groupings of objects (see [Sites \(EAM\) \[page 30\]](#)). The following types of objects can be attached to a site, can appear inside the site symbol, and be listed on the site's property sheet [Dependencies](#) tab. Objects can be attached to multiple sites, except where noted:
 - Organization units (see [Organization Units \(EAM\) \[page 32\]](#)).
 - [one site only] People (see [People \(EAM\) \[page 33\]](#)).
 - Business functions (see [Business Functions \(EAM\) \[page 48\]](#)).
 - Systems and databases (see [Systems, Applications, and Databases \(EAM\) \[page 58\]](#)).
 - [one site only] Hardware servers, workstations, mobile devices, and network nodes (see [Servers, Workstations, Mobile Devices, and Network Nodes \(EAM\) \[page 76\]](#)), and networks (see [Networks \(EAM\) \[page 78\]](#)).



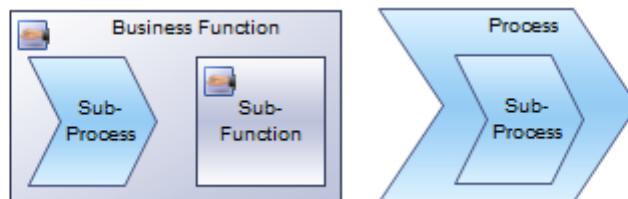
- Networks - Provide virtual groupings of objects (see [Networks \(EAM\) \[page 78\]](#)). Hardware objects and deployment instances can be attached to a network, can appear inside the network symbol, and be listed on various tabs of the network's property sheet.



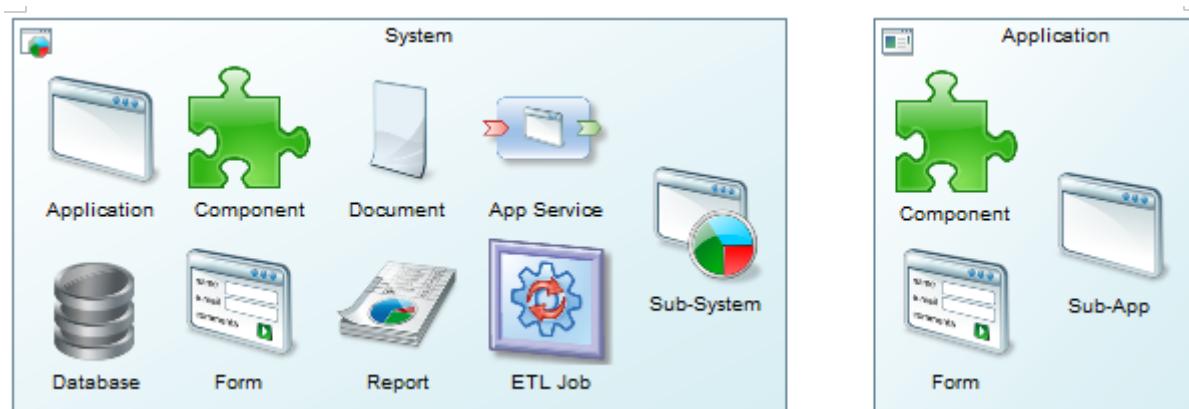
Note

Objects associated with an architecture area, site, or network do not belong to these objects, are not listed under them in the Browser, and will not be deleted if you delete the area, site or network. However, areas can contain sub-areas, sites can contain sub-sites, and networks can contain sub-networks, and these sub-objects do belong to their parents and will be deleted if their parent is deleted.

- Business functions can contain sub-functions and processes (see [Business Functions \(EAM\) \[page 48\]](#)), and processes can contain sub-processes (see [Processes \(EAM\) \[page 49\]](#)). Such sub-objects will be deleted if their parent is deleted.



- Systems and applications can contain other application architecture objects (see [Systems, Applications, and Databases \(EAM\) \[page 58\]](#)). When created or dragged inside a system or application, these sub-objects belong to the parent object, are listed under it in the Browser, and will be deleted if you delete the parent object:
 - Applications and databases
 - Components (see [Components \(EAM\) \[page 61\]](#))
 - In addition, systems only can contain the following sub-objects:
 - Sub-systems
 - Forms, documents, and reports (see [Forms, Documents, and Reports \(EAM\) \[page 62\]](#)).
 - ETL jobs (see [ETL Jobs \(EAM\) \[page 65\]](#)).
 - Application Services (see [Application and Business Services \(EAM\) \[page 68\]](#)).



- Servers, workstations, and mobile devices can have application architecture objects deployed to them (see [Servers, Workstations, Mobile Devices, and Network Nodes \(EAM\) \[page 76\]](#)). The deployment is modeled with deployment instances (see [Deployment Instances \(EAM\) \[page 80\]](#)). Each deployment instance is a sub-object of its parent hardware object, and will be deleted if you delete the parent, leaving the software asset itself unaffected.



1.4.2 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**.

In the **Display Preferences** dialog, select the type of object in the list in the left pane, and modify its appearance in the right pane.

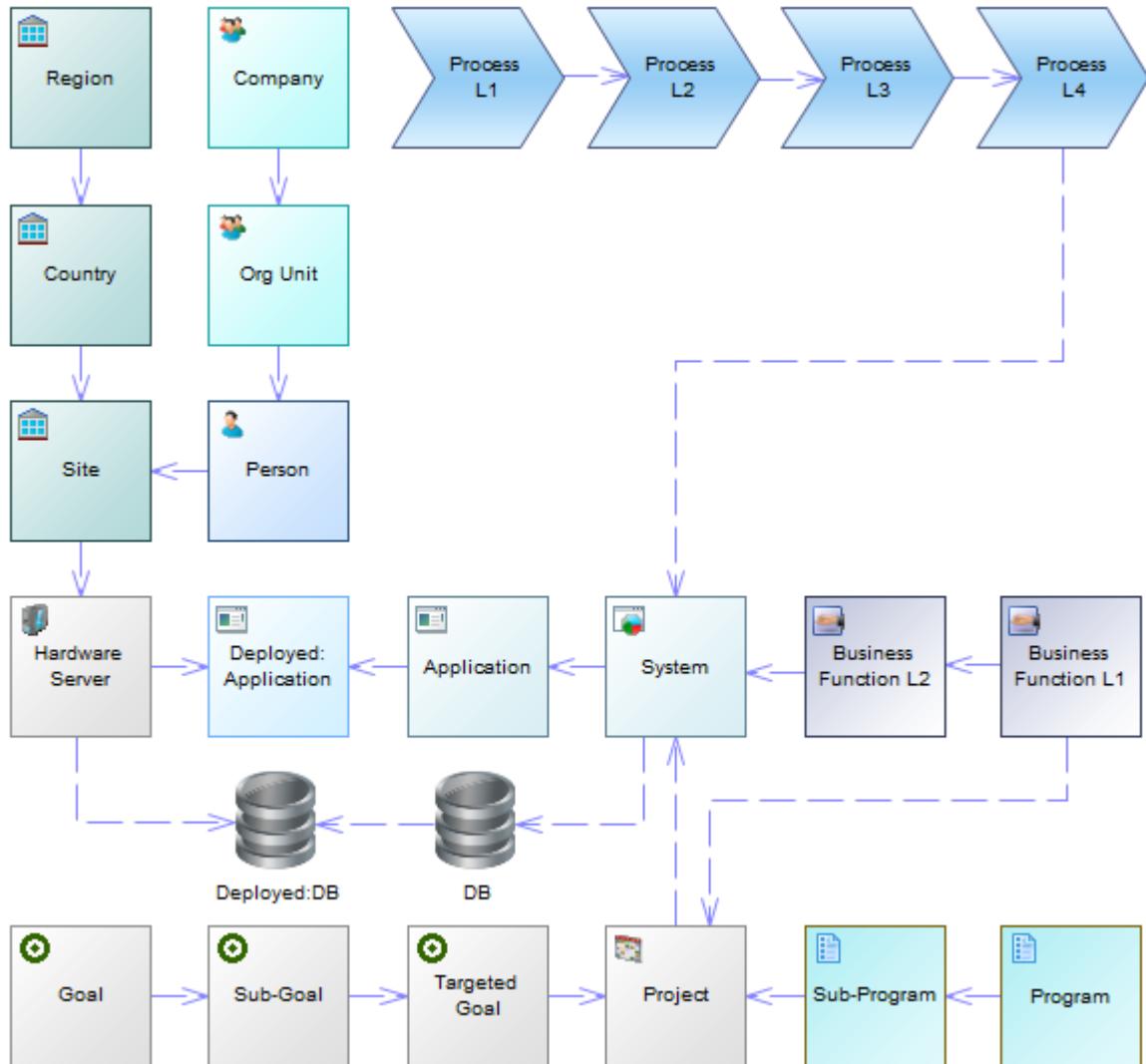
You can control what properties it will display on the **Content** tab, and how it will look on the **Format** tab. If the properties that you want to display are not available for selection on the **Content** tab, click the **Advanced** button and add them using the **Customize Content** dialog.

For detailed information about controlling the appearance and content of object symbols, see *Core Features Guide > Modeling with PowerDesigner > Diagrams, Matrices, and Symbols > Display Preferences*.

1.5 Example EA Model, Imports, and Charts

The EAM metamodel is very open and can be used to model your EA assets in many different structures. The `EAExample.eam` and its associated files are all based around one possible use of the EAM metamodel, and can be used as a starting point for your own choices.

The `EAExample.eam` EAM is structured as follows:



In this structure, only certain types of the available objects are selected to be modeled, and they are organized in a particular structure:

- In organization architecture (see [Organization Architecture Modeling \[page 27\]](#)):
 - Sites - are modeled in three levels, from regions, through countries, to sites.
 - Organization Units - are modeled in two levels, with people belonging to the second-level units and associated with a site.
- In process and function architecture (see [Process and Function Architecture Modeling \[page 43\]](#)):
 - Processes - are modeled in four levels, with fourth-level processes linked to systems via traceability links.
 - Business functions - are modeled in two levels, with first-level functions linked to projects and second-level functions linked to systems via traceability links.
- In application architecture (see [Application Architecture Modeling \[page 54\]](#)):
 - Systems can contain applications and databases.
- In infrastructure architecture (see [Infrastructure Architecture Modeling \[page 74\]](#)):
 - Systems are deployed to hardware servers, which are, in turn, associated with sites.
- In goal and project modeling (see [Goal and EA Project Modeling \[page 86\]](#)):
 - Projects are linked to sub-programs and programs.

- Goals are modeled in three levels, with third-level goals associated with projects.
 - Programs are modeled in two levels, with second-level programs containing projects.
 - Projects impact systems.
- In addition to the links shown in the diagram, people are linked via role associations (see [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)) to:
 - Business functions
 - Processes
 - Hardware servers
 - Databases
 - Applications
 - Goals
 - Programs
 - Projects

The EA Example model and its associated files are available in the `<install_dir>/Examples/EAExample` folder. The following files are provided:

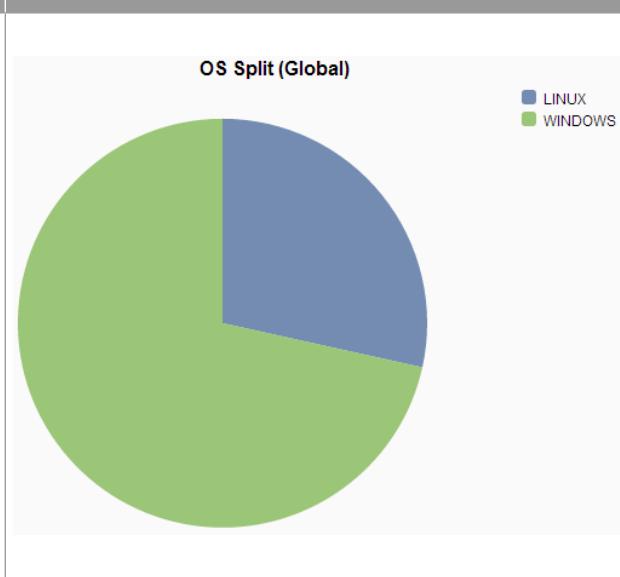
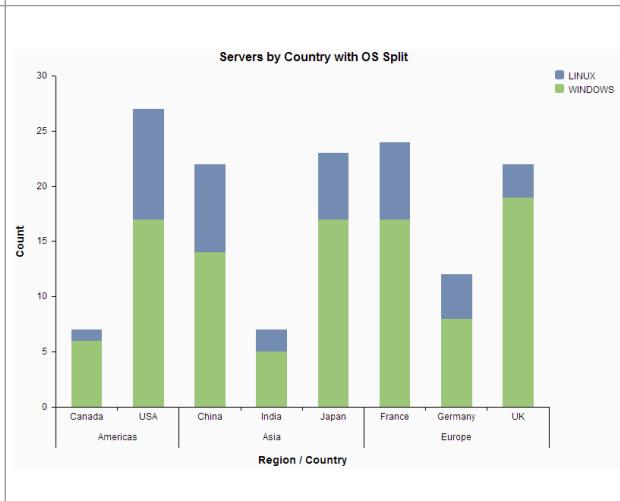
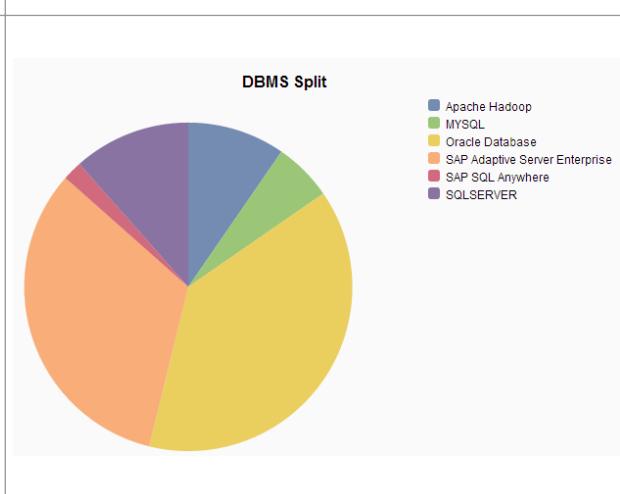
- A set of Excel files containing metadata to import (see [Importing EA Metadata from Excel \[page 126\]](#)).
- The `EAExample.eam` model, which is built from these Excel imports and contains additional diagrams (which cannot be imported through Excel).
- The `EAExample.xem` extension file, which contains dataset definitions to drive charts to analyze the metadata contained in the model (see [Chart Examples \[page 21\]](#)). This file is loaded by default in your repository `Library` folder and can be used directly with your own models if they conform to the `EAExample.eam` structure or be adapted to your chosen structure.
- The `EAExample.db` example SQL Anywhere repository database, which can be used to view the model and its charts using the PowerDesigner Web client. For information about installing and accessing this repository, see *Installation Guide > Installing the Repository > Installing the EA Example Repository*.

1.5.1 Chart Examples

Example models and an extension file containing datasets to derive charts from them are provided with PowerDesigner.

The following example charts are defined in the EA Charts extension file, which is loaded by default in your repository library (and which is also delivered as part of the EA Example files available at `<install_dir>/Examples/EAExample`). For information about working with datasets, see *Customizing and Extending PowerDesigner > Extension Files > Chart Datasets (Profile)*

Table 4:

Dataset and Chart Configuration	Chart																																				
<p>Question: What OS are my servers running?</p> <p>Path: Model > Hardware Server (Operating System, Operating System Version)</p> <p>Chart: A pie chart with each OS as a piece of the pie:</p> <ul style="list-style-type: none"> Pie Sectors: Count (of hardware servers) Legend Color: Operating System 	 <p>A pie chart titled "OS Split (Global)" showing the distribution of operating systems. The chart is divided into two main segments: a large green segment representing Windows and a smaller blue segment representing Linux.</p> <table border="1"> <caption>Data for OS Split (Global) Pie Chart</caption> <thead> <tr> <th>Operating System</th> <th>Count</th> </tr> </thead> <tbody> <tr> <td>LINUX</td> <td>~15</td> </tr> <tr> <td>WINDOWS</td> <td>~85</td> </tr> </tbody> </table>	Operating System	Count	LINUX	~15	WINDOWS	~85																														
Operating System	Count																																				
LINUX	~15																																				
WINDOWS	~85																																				
<p>Question: What OSs are running where?</p> <p>Path: Site > Site > Site > Hardware Server (Operating System, Operating System Version)</p> <p>Chart: A stacked bar chart with OSs as stacked colored bars</p> <ul style="list-style-type: none"> Measures: Count (of hardware servers) X Axis: Region, Country Legend Color: Operating System 	 <p>A stacked bar chart titled "Servers by Country with OS Split" showing the count of servers by region and country, broken down by operating system. The chart shows that the USA has the highest number of servers, followed by France and Japan.</p> <table border="1"> <caption>Data for Servers by Country with OS Split</caption> <thead> <tr> <th>Region / Country</th> <th>WINDOWS</th> <th>LINUX</th> <th>Total</th> </tr> </thead> <tbody> <tr> <td>Canada</td> <td>~6</td> <td>~1</td> <td>~7</td> </tr> <tr> <td>USA</td> <td>~18</td> <td>~8</td> <td>~26</td> </tr> <tr> <td>China</td> <td>~14</td> <td>~8</td> <td>~22</td> </tr> <tr> <td>India</td> <td>~5</td> <td>~2</td> <td>~7</td> </tr> <tr> <td>Japan</td> <td>~18</td> <td>~5</td> <td>~23</td> </tr> <tr> <td>France</td> <td>~18</td> <td>~6</td> <td>~24</td> </tr> <tr> <td>Germany</td> <td>~12</td> <td>~3</td> <td>~15</td> </tr> <tr> <td>UK</td> <td>~19</td> <td>~3</td> <td>~22</td> </tr> </tbody> </table>	Region / Country	WINDOWS	LINUX	Total	Canada	~6	~1	~7	USA	~18	~8	~26	China	~14	~8	~22	India	~5	~2	~7	Japan	~18	~5	~23	France	~18	~6	~24	Germany	~12	~3	~15	UK	~19	~3	~22
Region / Country	WINDOWS	LINUX	Total																																		
Canada	~6	~1	~7																																		
USA	~18	~8	~26																																		
China	~14	~8	~22																																		
India	~5	~2	~7																																		
Japan	~18	~5	~23																																		
France	~18	~6	~24																																		
Germany	~12	~3	~15																																		
UK	~19	~3	~22																																		
<p>Question: What DBMSs are my servers running?</p> <p>Path: Model > Database (DBMS, DBMS Version)</p> <p>Chart: A pie chart with each DBMS as a piece of the pie:</p> <ul style="list-style-type: none"> Pie Sectors: Count (of databases) Legend Color: DBMS 	 <p>A pie chart titled "DBMS Split" showing the distribution of database management systems. The chart is divided into several segments, with the largest being Oracle Database.</p> <table border="1"> <caption>Data for DBMS Split</caption> <thead> <tr> <th>DBMS</th> <th>Count</th> </tr> </thead> <tbody> <tr> <td>Oracle Database</td> <td>~45</td> </tr> <tr> <td>SAP Adaptive Server Enterprise</td> <td>~25</td> </tr> <tr> <td>SQLSERVER</td> <td>~10</td> </tr> <tr> <td>MySQL</td> <td>~5</td> </tr> <tr> <td>Apache Hadoop</td> <td>~5</td> </tr> <tr> <td>SAP SQL Anywhere</td> <td>~2</td> </tr> </tbody> </table>	DBMS	Count	Oracle Database	~45	SAP Adaptive Server Enterprise	~25	SQLSERVER	~10	MySQL	~5	Apache Hadoop	~5	SAP SQL Anywhere	~2																						
DBMS	Count																																				
Oracle Database	~45																																				
SAP Adaptive Server Enterprise	~25																																				
SQLSERVER	~10																																				
MySQL	~5																																				
Apache Hadoop	~5																																				
SAP SQL Anywhere	~2																																				

Dataset and Chart Configuration	Chart																																																						
<p>Question: What DBMSs are deployed where?</p> <p>Path: ► Site ► Site ► Site ► Hardware Server ► Deployment Instance ► Database (DBMS, DBMS Version) ▶</p> <p>Chart: A tree map:</p> <ul style="list-style-type: none"> • Area Weight: Count (of databases) • Area Color: Count (of databases) • Area Name: Region, Country, Site, DBMS 	<p>A treemap visualization showing database deployment across different sites. The main categories are Asia, Europe, and Americas. In Asia, there are sites like Shanghai, Guangzhou, Japan, Osaka, and India (Pune). In Europe, there are UK (Swindon), France (Marseilles), Germany (Frankfurt), and London. In the Americas, there are USA (New York, Boston, San Francisco, Canada (Toronto), and various Oracle, MySQL, SAP Adaptive Server, and SAP Hadoop instances.</p>																																																						
<p>Question: Who are my process owners and what processes do they own?</p> <p>Path: ► Organization Unit ► Organization Unit ► Person ► Role Association ► Process ▶</p> <p>Chart: A bar chart with a bar per person:</p> <ul style="list-style-type: none"> • Y Axis: Count (of processes) • X Axis: Person • Legend Color: Organization Unit 	<p>A bar chart titled "Number of Processes Owned by Owner". The Y-axis represents the count of processes (0 to 14), and the X-axis lists various process owners. The legend indicates organization units: Operations (blue), IT (green), Sales (yellow), HR (orange), Procurement (red), and Finance (purple). The chart shows that some individuals own many processes, while others own very few.</p> <table border="1"> <thead> <tr> <th>Process Owner</th> <th>Count</th> </tr> </thead> <tbody> <tr><td>Axel Danch</td><td>13</td></tr> <tr><td>Bordie Taylor</td><td>6</td></tr> <tr><td>Bruce Spence</td><td>4</td></tr> <tr><td>Chantal Freier</td><td>2</td></tr> <tr><td>Christophine Kinty</td><td>3</td></tr> <tr><td>Connieach Alexander</td><td>6</td></tr> <tr><td>David Bowers</td><td>6</td></tr> <tr><td>Dean Mitchell</td><td>3</td></tr> <tr><td>Dominique Chioncho</td><td>7</td></tr> <tr><td>Fay David</td><td>3</td></tr> <tr><td>Genevieve Reilly</td><td>2</td></tr> <tr><td>Goran Radic</td><td>1</td></tr> <tr><td>Hanshi Rotzburgh</td><td>5</td></tr> <tr><td>Jerome Blake</td><td>9</td></tr> <tr><td>Joel Edgerton</td><td>14</td></tr> <tr><td>John Knoll</td><td>9</td></tr> <tr><td>Karen Lucas</td><td>4</td></tr> <tr><td>Lawrence Foster</td><td>4</td></tr> <tr><td>Mark Sennell</td><td>8</td></tr> <tr><td>Matt Rowan</td><td>2</td></tr> <tr><td>Nina Fallon</td><td>3</td></tr> <tr><td>Pablo Hidalgo</td><td>1</td></tr> <tr><td>Rena Owen</td><td>1</td></tr> <tr><td>Robert Boundard</td><td>6</td></tr> <tr><td>Rohan Nichol</td><td>11</td></tr> <tr><td>Tim Gibbons</td><td>5</td></tr> </tbody> </table>	Process Owner	Count	Axel Danch	13	Bordie Taylor	6	Bruce Spence	4	Chantal Freier	2	Christophine Kinty	3	Connieach Alexander	6	David Bowers	6	Dean Mitchell	3	Dominique Chioncho	7	Fay David	3	Genevieve Reilly	2	Goran Radic	1	Hanshi Rotzburgh	5	Jerome Blake	9	Joel Edgerton	14	John Knoll	9	Karen Lucas	4	Lawrence Foster	4	Mark Sennell	8	Matt Rowan	2	Nina Fallon	3	Pablo Hidalgo	1	Rena Owen	1	Robert Boundard	6	Rohan Nichol	11	Tim Gibbons	5
Process Owner	Count																																																						
Axel Danch	13																																																						
Bordie Taylor	6																																																						
Bruce Spence	4																																																						
Chantal Freier	2																																																						
Christophine Kinty	3																																																						
Connieach Alexander	6																																																						
David Bowers	6																																																						
Dean Mitchell	3																																																						
Dominique Chioncho	7																																																						
Fay David	3																																																						
Genevieve Reilly	2																																																						
Goran Radic	1																																																						
Hanshi Rotzburgh	5																																																						
Jerome Blake	9																																																						
Joel Edgerton	14																																																						
John Knoll	9																																																						
Karen Lucas	4																																																						
Lawrence Foster	4																																																						
Mark Sennell	8																																																						
Matt Rowan	2																																																						
Nina Fallon	3																																																						
Pablo Hidalgo	1																																																						
Rena Owen	1																																																						
Robert Boundard	6																																																						
Rohan Nichol	11																																																						
Tim Gibbons	5																																																						
<p>Question: What is my IT Capex by Program and Project?</p> <p>Path: ► Program ► Program ► Project (ItCapex) ▶</p> <p>Chart: A tree map:</p> <ul style="list-style-type: none"> • Area Weight: ItCapex • Area Color: ItCapex • Area Name: Program, SubProgram, Project 	<p>A treemap visualization showing IT capital expenditure by program and project. The main categories are Project Destiny, Project Random, Project Nevermind, Project Vanity, Project Cord, Project Wall, Project Olive, Project Sandal, Project Gold, Project Zinc, Project Hand, and Project Iron. The size of each segment represents its ItCapex value, with a color gradient from light green (low) to dark green (high).</p>																																																						

Dataset and Chart Configuration	Chart																																
<p>Question: What is my IT Capex by Goal?</p> <p>Path: Goal Goal Goal Fulfillment Project (ItCapex) </p> <p>Chart: A pie chart with each goal as a piece of the pie:</p> <ul style="list-style-type: none"> Pie Sectors: ItCapex Legend Color: Goal 	<p>IT Capital Expenditure by Goal</p> <p>Legend:</p> <ul style="list-style-type: none"> Move CRM Services to the Cloud by end 2... Move HR systems to the Cloud by end 2015 Move Online Store to the Cloud by end 2017 Refactor Internal Code as Basis for Share... Update all DBMSs to Current-1 version wit... 																																
<p>Question: What is my IT Capex by Business Function?</p> <p>Path: BusinessFunction BusinessFunction TraceabilityLink Project (ItCapex) </p> <p>Chart: A bar chart with a bar per business function:</p> <ul style="list-style-type: none"> Y Axis: ItCapex X Axis: L1 Business Function 	<p>IT Capex by Business Function</p> <table border="1"> <thead> <tr> <th>Business Function</th> <th>ItCapex</th> </tr> </thead> <tbody> <tr><td>Analysis and Reporting</td><td>850k</td></tr> <tr><td>Enterprise Planning</td><td>250k</td></tr> <tr><td>Finance and Controlling</td><td>250k</td></tr> <tr><td>Human Resources</td><td>1050k</td></tr> <tr><td>IT Services</td><td>550k</td></tr> <tr><td>Marketing</td><td>80k</td></tr> <tr><td>Master Data Management</td><td>80k</td></tr> <tr><td>Production</td><td>1300k</td></tr> <tr><td>Sales</td><td>250k</td></tr> <tr><td>Storage and Distribution</td><td>100k</td></tr> <tr><td>Supply Chain Planning</td><td>600k</td></tr> </tbody> </table>	Business Function	ItCapex	Analysis and Reporting	850k	Enterprise Planning	250k	Finance and Controlling	250k	Human Resources	1050k	IT Services	550k	Marketing	80k	Master Data Management	80k	Production	1300k	Sales	250k	Storage and Distribution	100k	Supply Chain Planning	600k								
Business Function	ItCapex																																
Analysis and Reporting	850k																																
Enterprise Planning	250k																																
Finance and Controlling	250k																																
Human Resources	1050k																																
IT Services	550k																																
Marketing	80k																																
Master Data Management	80k																																
Production	1300k																																
Sales	250k																																
Storage and Distribution	100k																																
Supply Chain Planning	600k																																
<p>Question: How complex are the dependencies of my business functions on applications?</p> <p>Path: BusinessFunction BusinessFunction TraceabilityLink System EnterpriseApplication </p> <p>Chart: A bar chart with a bar per business function:</p> <ul style="list-style-type: none"> Y Axis: Count (of applications) X Axis: L1 Business Function 	<p>BusinessFunction App Dependencies</p> <table border="1"> <thead> <tr> <th>Business Function</th> <th>Count</th> </tr> </thead> <tbody> <tr><td>Analysis and Reporting</td><td>9</td></tr> <tr><td>Corporate Strategy</td><td>10</td></tr> <tr><td>Enterprise Planning</td><td>17</td></tr> <tr><td>Finance and Controlling</td><td>21</td></tr> <tr><td>Human Resources</td><td>7</td></tr> <tr><td>IT Services</td><td>10</td></tr> <tr><td>Legal</td><td>4</td></tr> <tr><td>Marketing</td><td>8</td></tr> <tr><td>Master Data Management</td><td>14</td></tr> <tr><td>Production</td><td>8</td></tr> <tr><td>Quality Management</td><td>4</td></tr> <tr><td>Research and Developme...</td><td>5</td></tr> <tr><td>Sales</td><td>8</td></tr> <tr><td>Storage and Distribution</td><td>10</td></tr> <tr><td>Supply Chain Planning</td><td>10</td></tr> </tbody> </table>	Business Function	Count	Analysis and Reporting	9	Corporate Strategy	10	Enterprise Planning	17	Finance and Controlling	21	Human Resources	7	IT Services	10	Legal	4	Marketing	8	Master Data Management	14	Production	8	Quality Management	4	Research and Developme...	5	Sales	8	Storage and Distribution	10	Supply Chain Planning	10
Business Function	Count																																
Analysis and Reporting	9																																
Corporate Strategy	10																																
Enterprise Planning	17																																
Finance and Controlling	21																																
Human Resources	7																																
IT Services	10																																
Legal	4																																
Marketing	8																																
Master Data Management	14																																
Production	8																																
Quality Management	4																																
Research and Developme...	5																																
Sales	8																																
Storage and Distribution	10																																
Supply Chain Planning	10																																

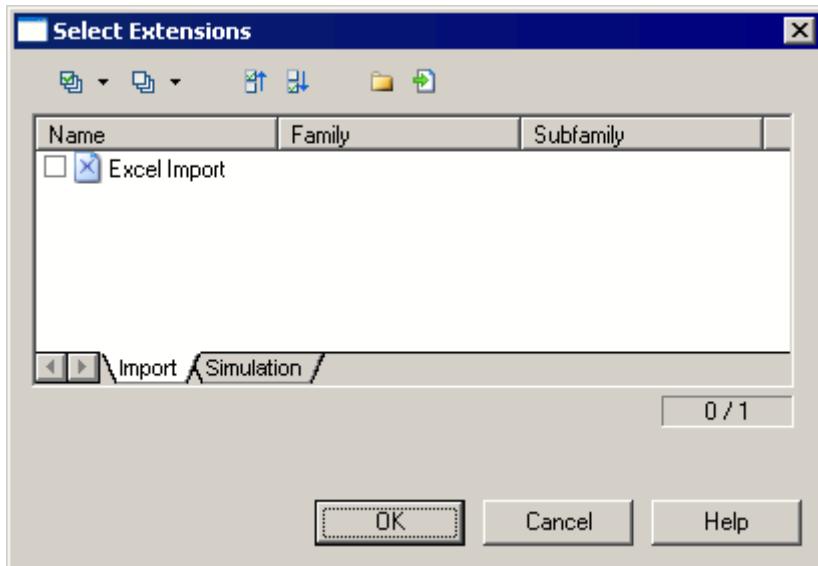
Dataset and Chart Configuration	Chart																
<p>Question: What is the impact of my projects on processes?</p> <p>Path: Process > Process > Process > Process > TraceabilityLink > System > Impact > Project</p> <p>Chart: A bar chart with a bar per business function:</p> <ul style="list-style-type: none"> Y Axis: Count (of projects) X Axis: L1 Business Function 	<p>The chart displays the count of projects across various business functions. The Y-axis represents the count, ranging from 0 to 600. The X-axis lists the business functions: Attract, Develop And Retain ..., Business Development Par..., Define & Track Strategy, Demand To Cash, Idea To Market, Procure To Pay, and Workplace & Infrastructure The bar for 'Procure To Pay' is the highest, exceeding 500.</p> <table border="1"> <thead> <tr> <th>Business Function</th> <th>Count</th> </tr> </thead> <tbody> <tr> <td>Attract, Develop And Retain ...</td> <td>~50</td> </tr> <tr> <td>Business Development Par...</td> <td>~20</td> </tr> <tr> <td>Define & Track Strategy</td> <td>~80</td> </tr> <tr> <td>Demand To Cash</td> <td>~90</td> </tr> <tr> <td>Idea To Market</td> <td>~50</td> </tr> <tr> <td>Procure To Pay</td> <td>> 500</td> </tr> <tr> <td>Workplace & Infrastructure ...</td> <td>~80</td> </tr> </tbody> </table>	Business Function	Count	Attract, Develop And Retain ...	~50	Business Development Par...	~20	Define & Track Strategy	~80	Demand To Cash	~90	Idea To Market	~50	Procure To Pay	> 500	Workplace & Infrastructure ...	~80
Business Function	Count																
Attract, Develop And Retain ...	~50																
Business Development Par...	~20																
Define & Track Strategy	~80																
Demand To Cash	~90																
Idea To Market	~50																
Procure To Pay	> 500																
Workplace & Infrastructure ...	~80																

1.6 Extending your Modeling Environment

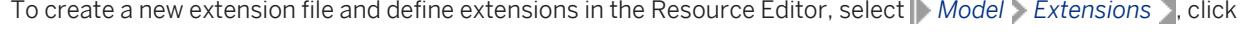
You can customize and extend PowerDesigner metaclasses, parameters, and file generation with extensions, which can be stored as part of your model or in separate extension files (*.xem) for reuse with other models.

To access extensions 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 quickly add a property or collection to an object from its property sheet, click the menu button in the bottom-left corner (or press F11) and select [New Attribute](#) or [New List of Associated Objects](#). For more information, see *Core Features Guide > Modeling with PowerDesigner > Objects > Extending Objects*.

To create a new extension file and define extensions in the Resource Editor, select  [Model](#)  [Extensions](#), click [Add a Row](#), and then click [Properties](#). For detailed information about working with extensions, see *Customizing and Extending PowerDesigner > Extension Files*.

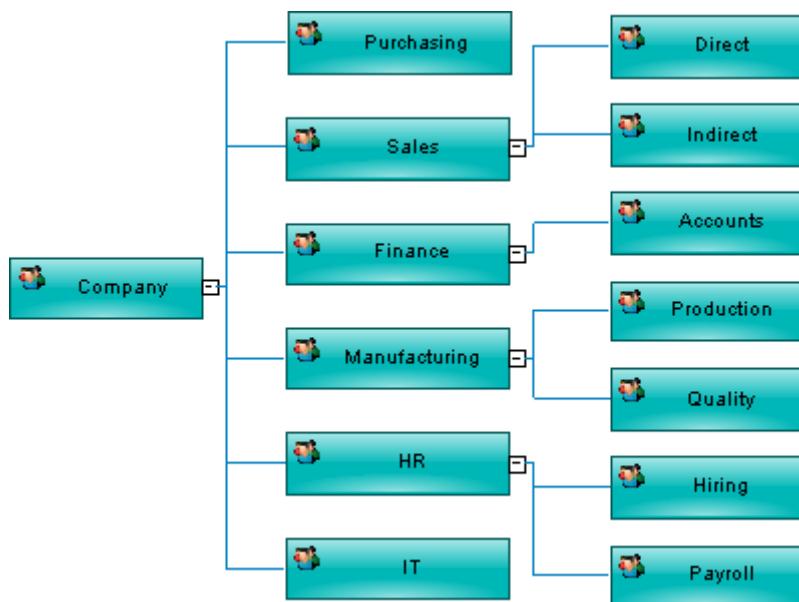
2 Organization Architecture Modeling

PowerDesigner provides you with various tools to capture, analyze, visualize, and plan changes to your organization. You can create organization charts to show the structure of your organization and business communication diagrams to show the information that transits across your sites and organization units.

2.1 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.

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



To create a 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**.

The following objects can be created from the organization chart toolbox:

Table 5:

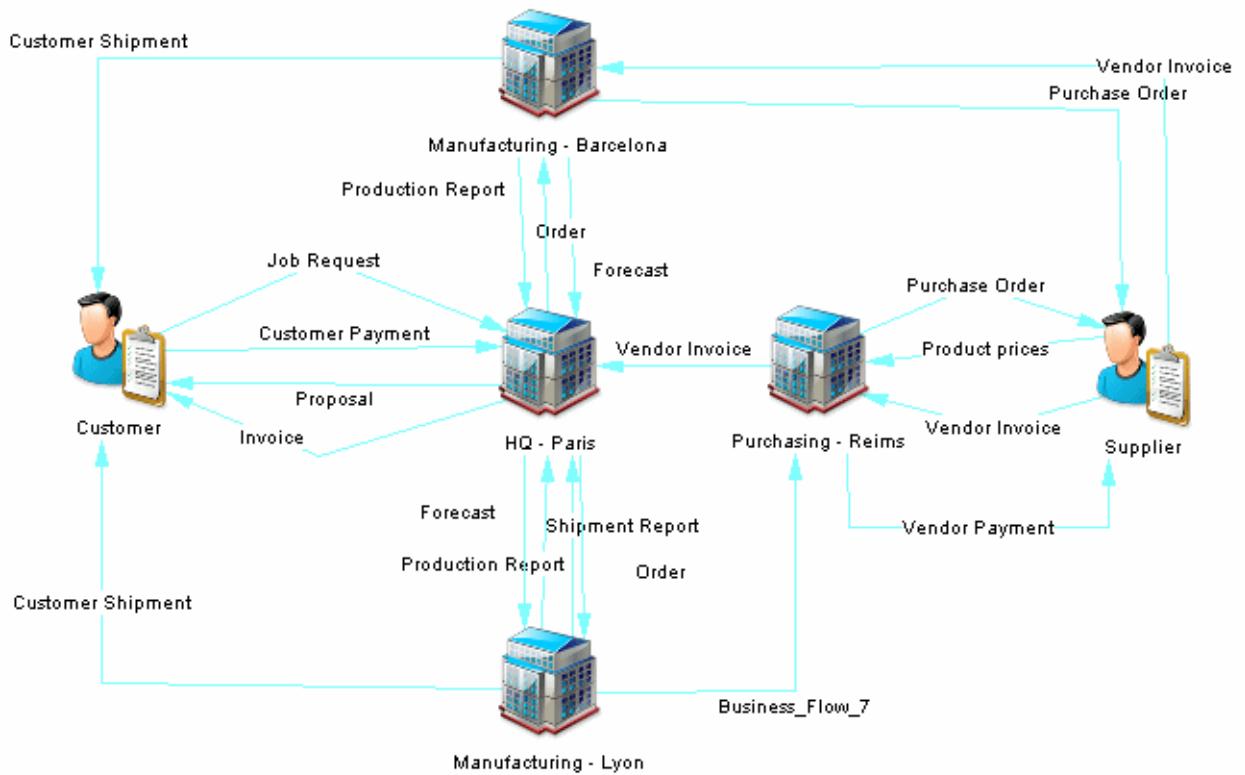
Tool	Description
	A group, department, function, or other collection of people or organization units. See Organization Units (EAM) [page 32] .
	An individual. See People (EAM) [page 33] .
	A link between organization units or people. See Roles (EAM) [page 39] .
	A high-level EA initiative. See Programs, Projects, and Phases (EAM) [page 91]
	An EA initiative. See Programs, Projects, and Phases (EAM) [page 91]
	A mission, vision, strategy, or objective. See Goals (EAM) [page 89]
	<i>Impact</i> - A link from an EA initiative to the asset that it impacts. See Specifying the Impact of a Project on EA Assets [page 93] .
	<i>Role Association</i> - A link from an EA asset to a person, group, or role that is responsible for or otherwise associated with it. See Associating a Person, Organization Unit, or Role with an Object [page 35] .

For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

2.2 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.

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



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**.

All objects can be displayed in a business communication diagram but the primary focus is on the following business objects, which can be created from the toolbox:

Table 6:

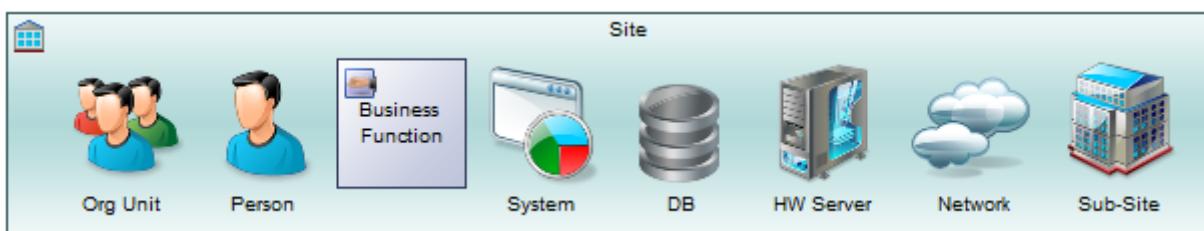
Tool	Description
	<i>Architecture Area</i> - An abstract object for grouping other objects. See Architecture Areas (EAM) [page 46] .
	<i>Business Function</i> - An aggregation of processes and/or sub-functions. See Business Functions (EAM) [page 48] .
	<i>Business Flow</i> - A link between elements in a business diagram. See Business Flows (EAM) [page 40] .
	<i>Process</i> - An activity or group of sub-processes. See Processes (EAM) [page 49] .
	<i>Site</i> - A physical location. See Sites (EAM) [page 30] .
	<i>Organization Unit</i> - A group, department, function, or other collection of people or organization units. See Organization Units (EAM) [page 32] .

Tool	Description
	<i>Person</i> - An individual. See People (EAM) [page 33] .
	<i>Role</i> - A set of responsibilities. See Roles (EAM) [page 39] .
	<i>Program</i> - A high-level EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Project</i> - An EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Goal</i> - A mission, vision, strategy, or objective. See Goals (EAM) [page 89] .
	<i>Impact</i> - A link from an EA initiative to the asset that it impacts. See Specifying the Impact of a Project on EA Assets [page 93] .
	<i>Role Association</i> - A link from an EA asset to a person, group, or role that is responsible for or otherwise associated with it. See Associating a Person, Organization Unit, or Role with an Object [page 35] .
	<i>Fulfillment</i> - A link from an EA initiative to a goal that it contributes to fulfilling. See Specifying the Fulfillment of Goals by a Project [page 92] .

For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

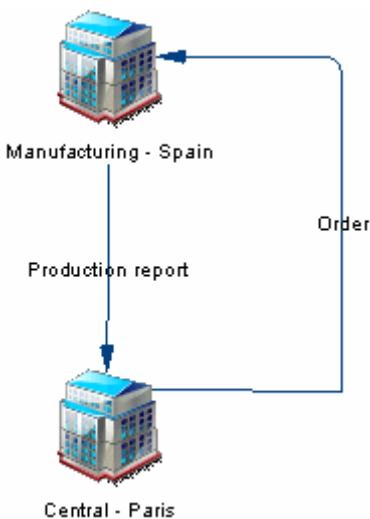
2.3 Sites (EAM)

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



For more information about grouping and organizing EAM objects, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

In this example, the Central and Manufacturing sites interact:



Creating a Site

You can create a site from the Toolbox, Browser, or *Model* menu, or on the *Sub-Sites* tab of another site. Sites can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 7:

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 <i>Code</i> 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.
Address / Zip code / City / Country	Specify the address of the site.
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.

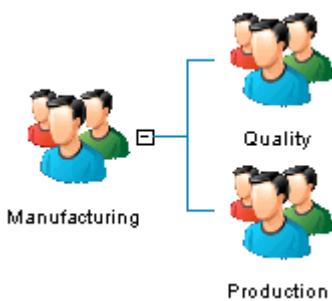
The following tabs are also available:

- 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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Sub-Sites – lists the sites contained within the site.

2.4 Organization Units (EAM)

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

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, or on the *Sub-Organizations* tab of another organization unit. Organization units can be displayed in the following diagrams:

- Organization chart (see [Organization Charts \[page 27\]](#))
- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))

- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 8:

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 <i>Code</i> 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.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

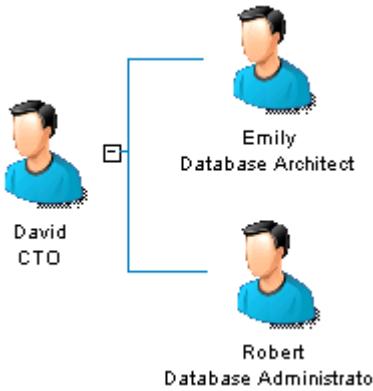
The following tabs are also available:

- Sub-Organizations – lists the organization units that belong to this organization unit.
- People – lists the people (see [People \(EAM\) \[page 33\]](#)) that are associated with this organization unit.
- Roles – lists the roles (see [Roles \(EAM\) \[page 39\]](#)) that are associated with this organization unit.
- Sites – lists the sites (see [Sites \(EAM\) \[page 30\]](#)) where this organization unit is located.

2.5 People (EAM)

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

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, or on the *People* tab of an organization unit. People can be displayed in the following diagrams:

- Organization chart (see [Organization Charts \[page 27\]](#))
- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 9:

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 <i>Code</i> 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
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 \(EAM\) \[page 39\]](#)) that this person fulfills.

2.5.1 Associating a Person, Organization Unit, or Role with an Object

You can link a person, organization unit, or role to another object using the Toolbox [Role Association](#) tool or on the [Role Associations](#) tab of the property sheet of the object. The role association link is displayed in the diagram, and the list of role associations for a person, organization unit, or role is available from the [Dependencies](#) tab of its property sheet.

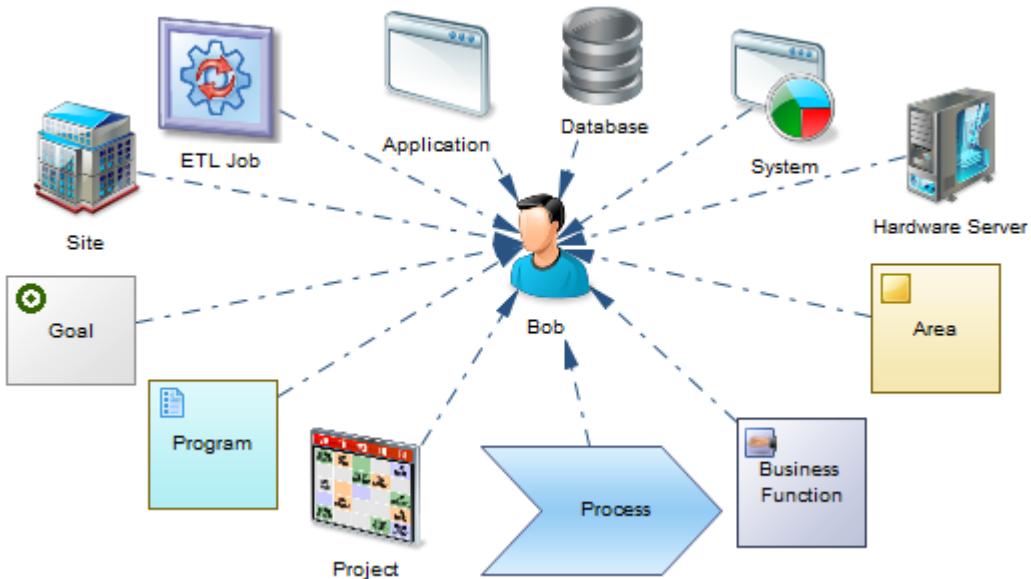
Context

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.

To create the role association directly in an organization chart, business communication diagram, or service-oriented diagram, click the [Role Association](#) tool in the Toolbox, and draw a link from the object to the person, organization unit, or role. Double-click the object to open its property sheet, click the [Role Associations](#) tab, and specify the role type.

In this example, Bob is responsible for a wide range of objects:



To specify role associations in any diagram, including those where the *Role Association* tool is not available:

Procedure

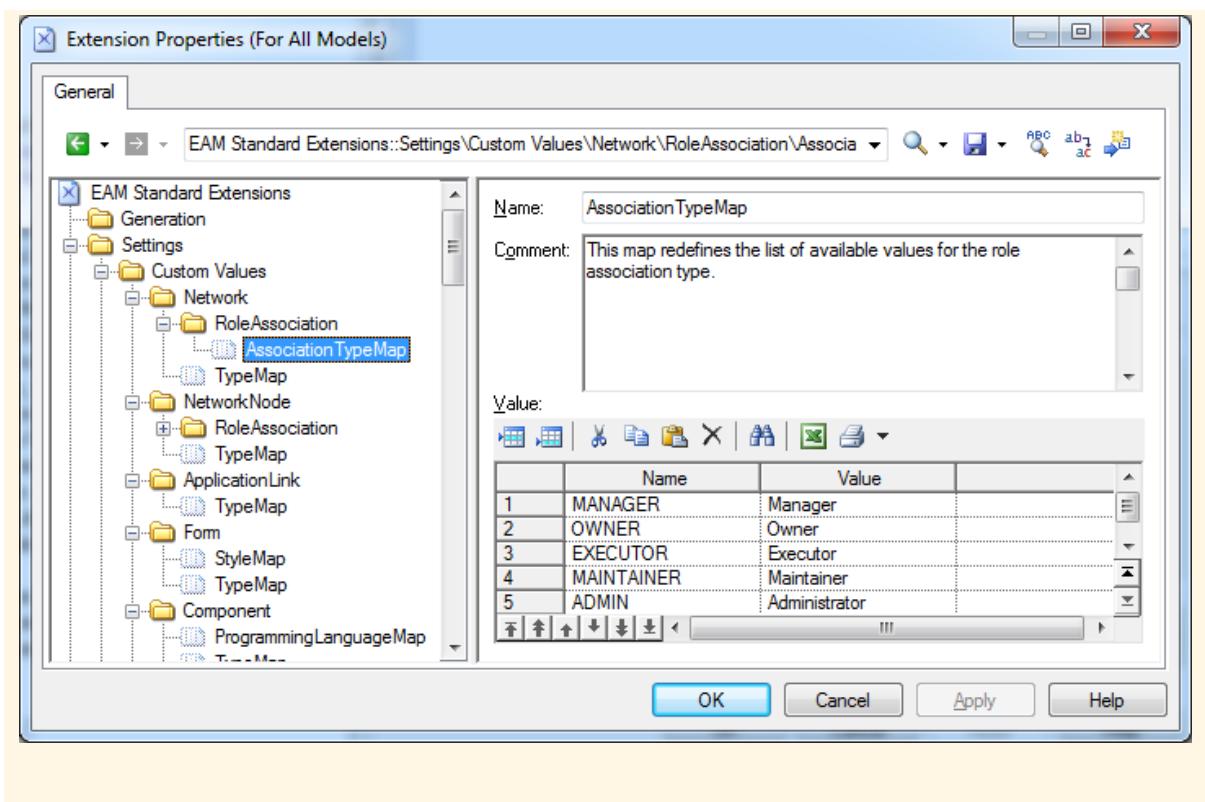
1. Open the property sheet of the object to which you want to connect a person, organization unit, or role, and click the *Role Associations* tab.
2. Click the *Add Objects* tool to open a selection box listing all the roles, people, and organization units available in the model, select the roles, people, and organization units that you want to associate with the object and click *OK* to return to the *Role Associations* tab.
3. [optional] Select or enter a type for each role association and then click *OK* to return to the diagram.
If the role player and the object are in the current diagram then the link will be drawn between them.

i Note

To display objects in a diagram, select **Symbol > Show Symbols** and select the relevant objects from the dialog. To display role associations, and other links in the diagram, select **Tools > Complete Links**.

i Note

The default list of types for each type of object can be modified in an extension file. Navigate to **Settings / Custom Values / <object>/RoleAssociation/AssociationTypeMap** and modify the list as appropriate:



2.5.2 Importing Role Associations from Excel

You can import role associations, which are used to link people to other assets in your models, from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

- To create a new EAM to import into, select **File > Import > Excel File**. Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting [Import Excel File](#). If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the [Select File](#) tool, browse to and select the Excel file to import, and click [Open](#) to return to the wizard.

The example file `05_Role Associations Import.xlsx` contains five sheets, each of which designates people as owners of various objects imported from precedent imports.

Note

Certain metadata in the file cannot be automatically evaluated by the wizard. This file cannot be imported using the [Auto-map columns to properties](#) option and you must specify the mappings in the wizard.

3. Click the [Options](#) button set the import options as follows, and click [OK](#) to return to the wizard:

- All options - deselected.
- [Reference associated object by: Code](#)
- [Qualified name separator: . \(dot\)](#)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

4. Click [Next](#) to begin the import.

Specify how your sheets and columns will be imported in the wizard as follows:

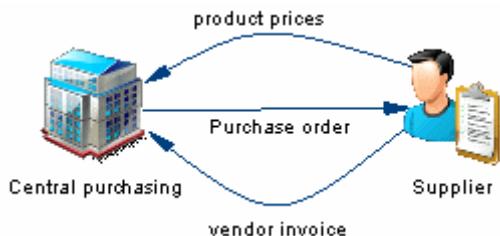
- Process Owners:**
 1. Import the table as **Process.Role Association**.
 2. Import the column **Parent** as **Parent**.
 3. Import the column **Role Player** as **Role Player** and set the **Object** type to **Person**.
 4. Import the column **Type** as **Type**.
- Function Owners** - Import the table as **Business Function.Role Association** and its columns in the same way as for **Process Owners**.
- Server Owners** - Import the table as **Hardware Server.Role Association** and its columns in the same way as for **Process Owners**.
- Application Owners** - Import the table as **Application.Role Association** and its columns in the same way as for **Process Owners**.
- Database Owners** - Import the table as **Database.Role Association** and its columns in the same way as for **Process Owners**.

Progress is displayed in the [Output](#) window. When the import is complete, a dialog will appear showing how many objects have been created. Click [OK](#) to return to your model and review the imported objects.

2.6 Roles (EAM)

A role is set of responsibilities. Roles can be assigned to people or organization units or can be used in place of them.

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. Roles can be displayed in the following diagrams:

- Organization chart (see [Organization Charts \[page 27\]](#))
- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 10:

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 <i>Code</i> 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.

2.7 Business Flows (EAM)

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

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. Business flows can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))

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:

Table 11:

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 <i>Code</i> 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.
First/ Second object	Specify the objects at the origin and endpoint 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 \(EAM\) \[page 62\]](#)).
- 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.

2.8 Importing Organization Architecture Metadata from Excel

You can import your site and organization structure and the people necessary for your models from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

1. To create a new EAM to import into, select Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting [Import Excel File](#). If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the [Select File](#) tool, browse to and select the Excel file to import, and click [Open](#) to return to the wizard.

The example file 01_Organization_Architecture_Import.xlsx contains three sheets:

- **Site** - Imports a three-level structure of sites (regions, countries, and city sites). The **Parent** column contains the code of the site to which this site belongs so that, for example, the site France (FR) is a child of the site Europe (EU), and the site Paris (PA) is a child of the site France (EU.FR). The full path to the site is stored in the **Comment**.
- **OrganizationUnit** - Imports a two-level structure of organization units, where the various divisions all belong to a single **Parent**.
- **Person** - Imports a number of people, each of whom is assigned to an organization unit, a manager, and a site.

Note

As all the necessary metadata is contained within the file, selecting the [Auto-map columns to properties](#) allows the wizard to import all the objects without further intervention.

3. Click the [Options](#) button set the import options as follows, and click [OK](#) to return to the wizard:

- [Auto-map columns to properties](#) (selected)
- All other options - deselected.
- [Reference associated object by: Code](#)
- [Qualified name separator: .](#) (dot)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

4. Click [Next](#) to begin the import.

Progress is displayed in the [Output](#) window. When the import is complete, a dialog will appear showing how many objects have been created. Click [OK](#) to return to your model and review the imported objects.

3 Process and Function Architecture Modeling

PowerDesigner lets you capture, analyze, visualize, and plan changes to your processes and functions. You can create process maps to show the hierarchy of your business processes and capabilities and city planning diagrams to give a big picture view of your business functions and architecture areas.

3.1 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.

The following example shows a top-level process map:



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**.

The following objects can be created from the process map toolbox:

Table 12:

Tool	Description
	<i>Process</i> - An activity or group of sub-processes. See Processes (EAM) [page 49] .
	<i>Business Function</i> - An aggregation of processes and/or sub-functions. See Business Functions (EAM) [page 48] .
	<i>Architecture Area</i> - An abstract object for grouping other objects. See Architecture Areas (EAM) [page 46] .
	<i>Goal</i> - A mission, vision, strategy, or objective. See Goals (EAM) [page 89] .
	<i>Program</i> - A high-level EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Project</i> - An EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Impact</i> - A link from an EA initiative to the asset that it impacts. See Specifying the Impact of a Project on EA Assets [page 93] .
	<i>Fulfillment</i> - A link from an EA initiative to a goal that it contributes to fulfilling. See Specifying the Fulfillment of Goals by a Project [page 92] .

For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

3.1.1 Creating a Multi-Level Process Map

Process maps are commonly maintained by process analysts who, starting from a top-level overview of business areas and high-level functions, decompose top-level processes into sub-processes. Some or all of the processes in the top-level map are decomposed into sub-processes containing sub-maps and so on down through a number of levels. Architecture areas and business functions are only permitted in the top-level and cannot be created in sub-maps.

It is common practice to decompose processes to four levels in a process map, and then to model the steps of each fourth-level process in a business process diagram (see [Linking Processes to Business Process Diagrams \[page 51\]](#)).

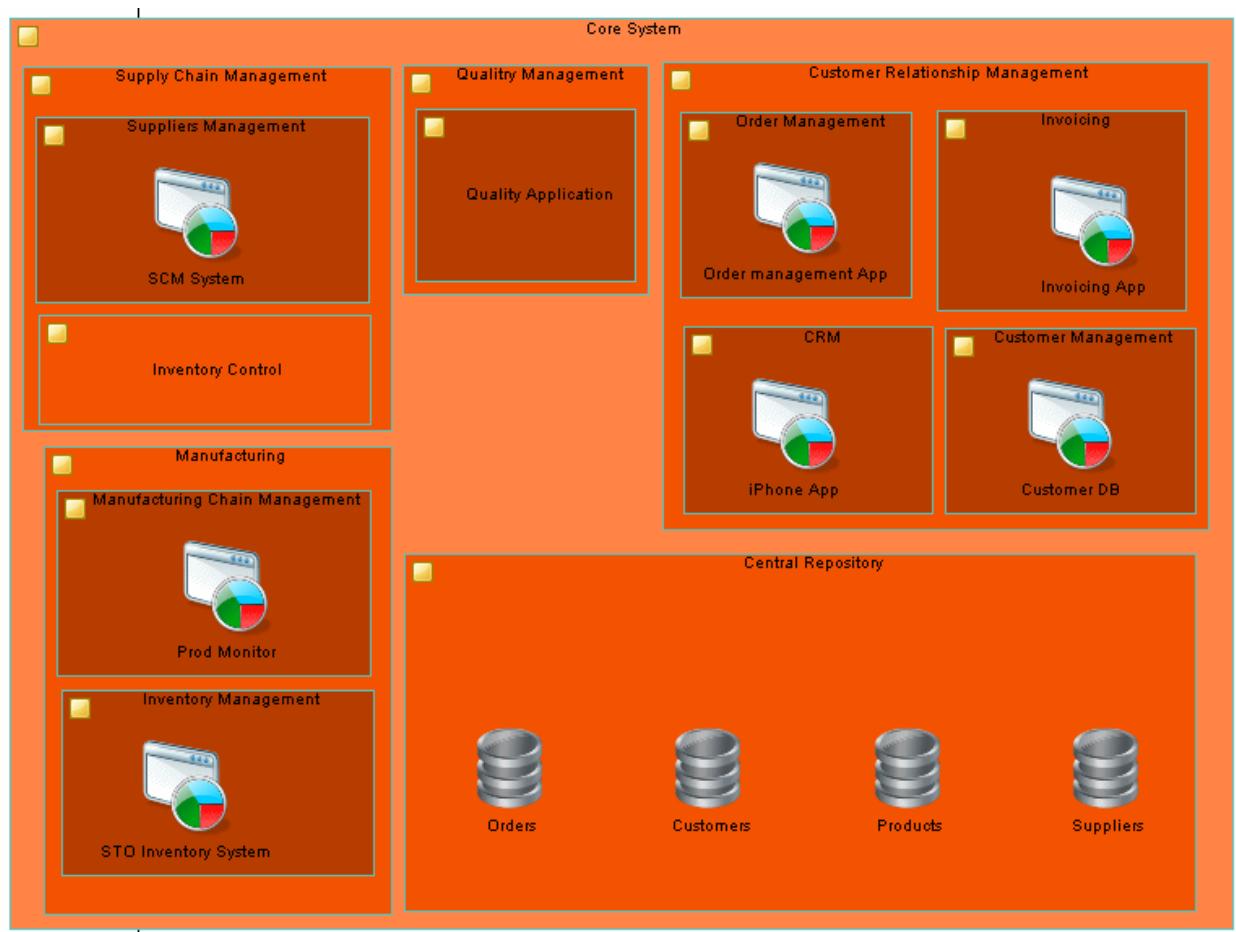
To create a sub-map inside a process (or to go down into a sub-map that has already been created, press **Ctrl** and double-click the process symbol. Processes that you create in this sub-map are sub-processes of the initial process, and are displayed under it in the Browser and on the *Processes* tab of its property sheet. To go back up to the parent map, press **Ctrl+U**.

Business functions can contain sub-functions and processes, but they cannot contain sub-maps. To create sub-functions or processes inside a function, use the *Business Function* or *Process* tool and click in an existing business function symbol. Sub-functions and processes created inside a business function are displayed under it in the Browser and on the *Sub-Functions* or *Processes* tab of its property sheet.

3.2 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 functions, systems, applications, etc into architectural areas.

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.



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**.

The following objects can be created from the city planning diagram toolbox:

Table 13:

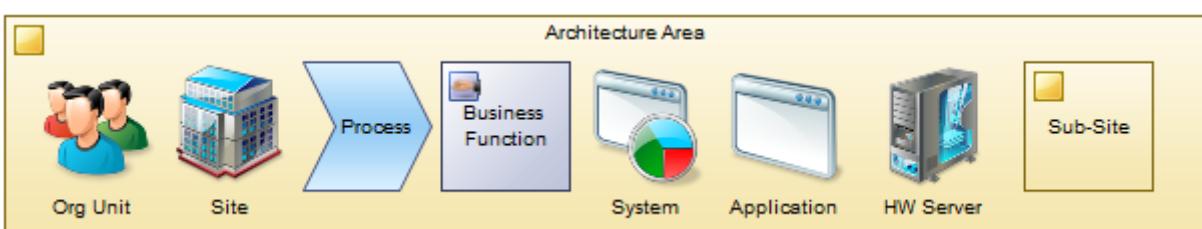
Tool	Description
	<i>Architecture Area</i> - An abstract object for grouping other objects. See Architecture Areas (EAM) [page 46] .

Tool	Description
	<i>System</i> - A packaged application. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Application</i> - A computer program. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Database</i> - A database. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Business Function</i> - An aggregation of processes and/or sub-functions. See Business Functions (EAM) [page 48] .
	<i>Program</i> - A high-level EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Project</i> - An EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Impact</i> - A link from an EA initiative to the asset that it impacts. See Specifying the Impact of a Project on EA Assets [page 93] .

For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

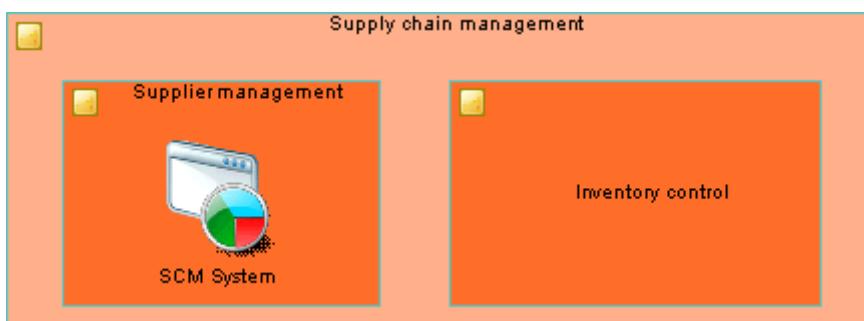
3.3 Architecture Areas (EAM)

An *architecture area* is an abstract object that can group together other objects. The objects do not belong to the area and are just grouped in it.



For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

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, or on the *Sub-Areas* tab of an architecture area. Architecture areas can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Process map (see [Process Maps \[page 43\]](#))
- City planning diagram (see [City Planning Diagrams \[page 45\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 14:

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 <i>Code</i> 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.

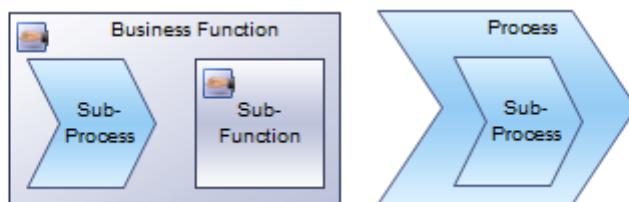
The following tabs are also available:

- 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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Attached Objects – lists the objects that are associated with the area.
- Sub-Areas – lists the architecture areas contained within the area.

3.4 Business Functions (EAM)

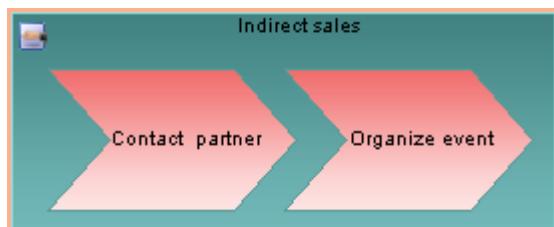
A *business function* is an aggregation of sub-functions and processes. These sub-objects belong to the function and will be deleted if you delete it.

Business functions can contain sub-functions and processes:



For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

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, or on the *Sub-Functions* tab of a business function. Business functions can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))

- Process map (see [Process Maps \[page 43\]](#))
- City planning diagram (see [City Planning Diagrams \[page 45\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))

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:

Table 15:

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 <i>Code</i> 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.

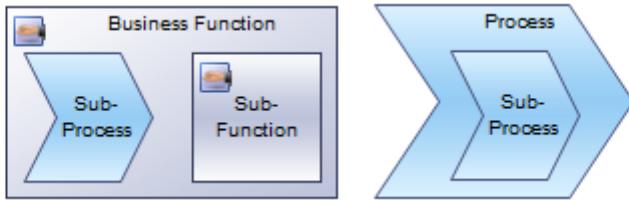
The following tabs are also available:

- 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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Sub-Functions – lists the business functions contained within the function.
- Processes – lists the processes (see [Processes \(EAM\) \[page 49\]](#)) contained within the function.
- Sites – lists the sites (see [Sites \(EAM\) \[page 30\]](#)) associated with the function.

3.5 Processes (EAM)

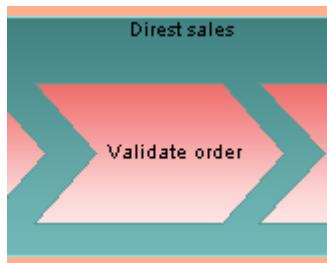
A process represents something that is done by the organization. Processes can be broken down into sub-processes and the hierarchy of processes is commonly shown in a process map.

Processes can be contained within business functions and can contain sub-processes:



For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

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, or on the *Processes* tab of a business function or the *Sub-Processes* tab of a process. Processes can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Process map (see [Process Maps \[page 43\]](#))
- City planning diagram (see [City Planning Diagrams \[page 45\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 16:

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 <i>Code</i> 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.
Number ID	Specifies the number of the process in the sequence. Numbers are allotted sequentially as processes are created. If you move processes around, the numbers may no longer correspond to the desired order. You can modify the number of a process here, and the other numbers will be changed to avoid duplications and fill holes in the sequence where possible.
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.

Sub-processes can appear inside the process symbol or in a sub-map, and are listed in both cases on the *Children* facet of its property sheet.

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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Sub-Processes - lists the child processes contained within the process. Use the *Add a Row* tool on this tab to create a new process.

3.5.1 Linking Processes to Business Process Diagrams

While you can decompose processes into sub-processes in a process map, it is common practice to model the steps of lower-level processes in a business process diagram. The process map is commonly maintained by an enterprise architect or process analyst, while the modeling of fourth-level process steps is often done by process owners.

Context

Note

You must create the business process diagram before you can link it to a process in your process map.

Procedure

1. Open the business process model containing the diagram to link to in your PowerDesigner desktop client workspace.
2. In the EAM containing your process map, open the property sheet of the process and click the *Related Diagrams* tab.
3. Click the *Add Objects* tool and, in the dialog, select the BPM in the *Models* list.
4. Select the diagram that you want to link to and then click *OK*.

The business process diagram is now associated with the process. You can navigate to it from the EAM process by pressing *Ctrl* and double-clicking the process symbol.

3.6 Importing Process Architecture Metadata from Excel

You can import your processes, business functions, and architectural areas from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA_Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

1. To create a new EAM to import into, select *File* *Import* *Excel File*. Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click *OK* to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting *Import Excel File*. If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the *Select File* tool, browse to and select the Excel file to import, and click *Open* to return to the wizard.

The example file 02_Process_Architecture_Import.xlsx contains three sheets:

- **Process** - Imports a four-level structure of processes to represent a typical process map. The **Parent** column contains the code of the process to which the process belongs. When working with a hierarchy of objects, you must use the full path to the object when referencing it from another object. In the following example, we show the values for the **Parent**, **Code**, and **Name** columns for a hierarchy of processes:

Table 17:

Process Level	Parent	Code	Name	Full Path
1	(none)	P1	Define & Track Strategy	P1
2	P1	P1_1	Manage Corporate Strategy	P1.P1_1
3	P1.P1_1	P1_1_4	Manage Product Strategy	P1.P1_1.P1_1_4
4	P1.P1_1.P1_1_4	P1_1_4_1	Manage Physical Assets	P1.P1_1.P1_1_4.P1_1_4_1

- **BusinessFunction** - Imports a two-level structure of business functions, which are used to represent capabilities.
- **ArchitectureArea** - Imports three abstract areas that can be used to loosely regroup objects into the **Management**, **Core**, and **Support** areas.

i Note

As all the necessary metadata is contained within the file, selecting the *Auto-map columns to properties* allows the wizard to import all the objects without further intervention.

3. Click the *Options* button set the import options as follows, and click *OK* to return to the wizard:

- *Auto-map columns to properties* (selected)
- All other options - deselected.
- *Reference associated object by: Code*
- *Qualified name separator: .* (dot)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

4. Click *Next* to begin the import.

Progress is displayed in the *Output* window. When the import is complete, a dialog will appear showing how many objects have been created. Click *OK* to return to your model and review the imported objects.

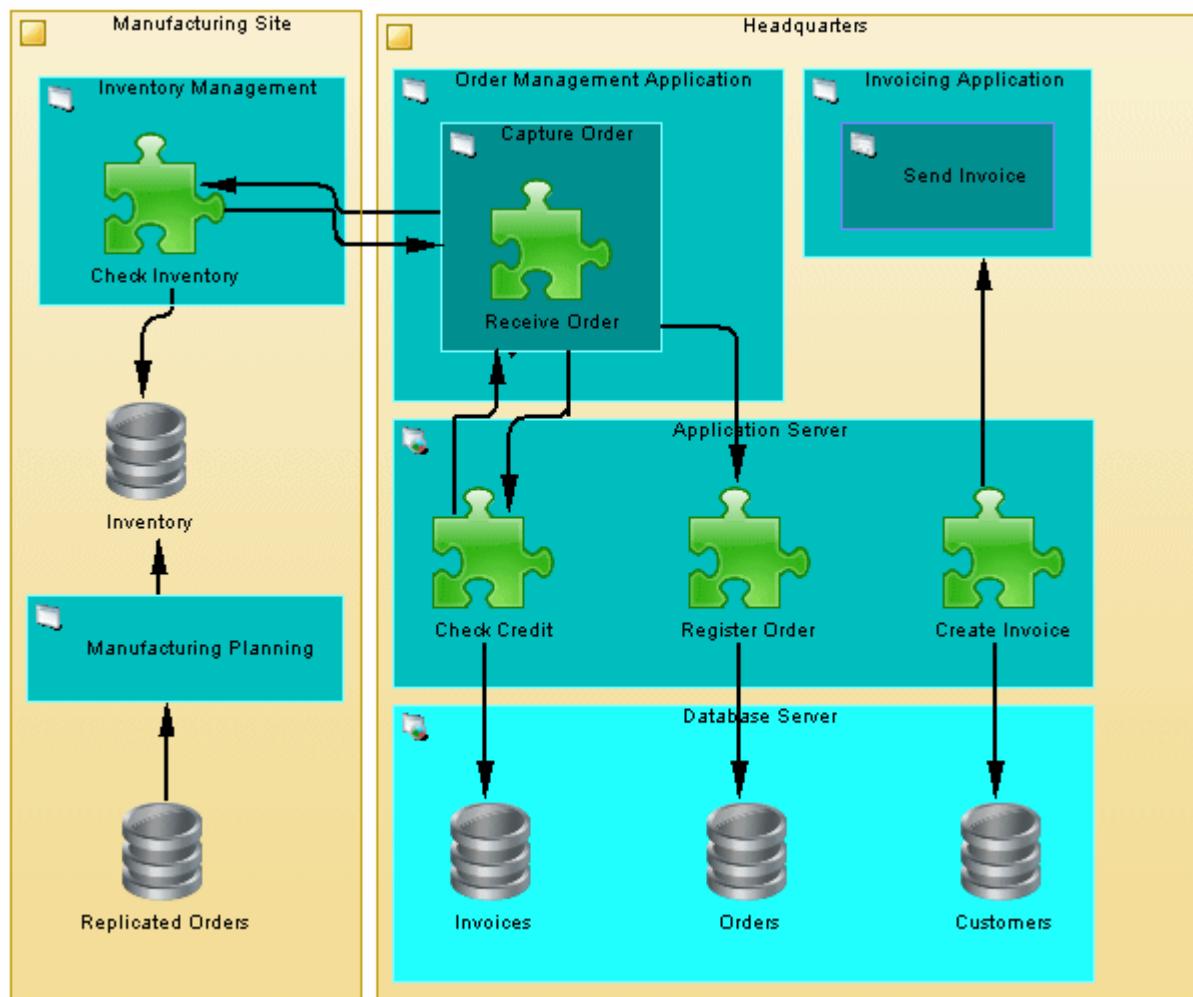
4 Application Architecture Modeling

PowerDesigner lets you capture, analyze, visualize, and plan changes to your applications, systems, databases, and services. You can create application architecture diagrams to identify the interactions between your applications and components and service-oriented diagrams to organize applications and services in terms of SOA layers.

4.1 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.

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



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**.

All objects can be displayed in an application architecture diagram but the primary focus is on the following software objects, which can be created from the toolbox:

Table 18:

Tool	Description
	<i>Architecture Area</i> - An abstract object for grouping other objects. See Architecture Areas (EAM) [page 46] .
	<i>Site</i> - A physical location. See Sites (EAM) [page 30] .
	<i>Application Service</i> - An externally visible unit of functionality. See Application and Business Services (EAM) [page 68] .
	<i>System</i> - A packaged application. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Application</i> - A computer program. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Database</i> - A database. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Component</i> - A replaceable part of an application. See Components (EAM) [page 61] .
	<i>Form</i> - A UI component. See Forms, Documents, and Reports (EAM) [page 62] .
	<i>Document</i> - A document. See Forms, Documents, and Reports (EAM) [page 62] .
	<i>Report Document</i> - A report. See Forms, Documents, and Reports (EAM) [page 62] .
	<i>ETL Job</i> - A data extraction, transformation, and load. See ETL Jobs (EAM) [page 65] .
	<i>Application Link</i> - A link between elements in an application layer diagram. See Application Links (EAM) [page 71] .
	<i>Program</i> - A high-level EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Project</i> - An EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Impact</i> - A link from an EA initiative to the asset that it impacts. See Specifying the Impact of a Project on EA Assets [page 93] .

For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

4.2 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.

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:



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**.

The following objects can be created from the service-oriented diagram toolbox:

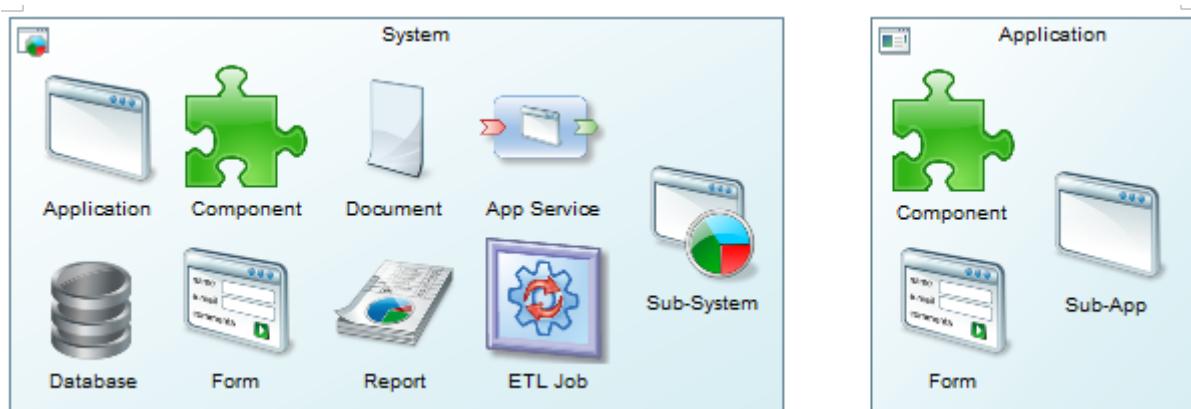
Table 19:

Tool	Description
	<i>Architecture Area</i> - An abstract object for grouping other objects. See Architecture Areas (EAM) [page 46] .
	<i>Role</i> - A set of responsibilities. See Roles (EAM) [page 39] .
	<i>Process</i> - An activity or group of sub-processes. See Processes (EAM) [page 49] .
	<i>Application Service</i> - An externally visible unit of functionality. See Application and Business Services (EAM) [page 68] .
	<i>Business Service</i> - A service offered by an organization. See Application and Business Services (EAM) [page 68] .
	<i>System</i> - A packaged application. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Application</i> - A computer program. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Database</i> - A database. See Systems, Applications, and Databases (EAM) [page 58] .
	<i>Component</i> - A replaceable part of an application. See Components (EAM) [page 61] .
	<i>Contract</i> - An agreement between services. See Contracts (EAM) [page 66] .
	<i>Application Link</i> - A link between elements in an application layer diagram. See Application Links (EAM) [page 71] .
	<i>Program</i> - A high-level EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Project</i> - An EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Impact</i> - A link from an EA initiative to the asset that it impacts. See Specifying the Impact of a Project on EA Assets [page 93] .
	<i>Role Association</i> - A link from an EA asset to a person, group, or role that is responsible for or otherwise associated with it. See Associating a Person, Organization Unit, or Role with an Object [page 35] .

For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

4.3 Systems, Applications, and Databases (EAM)

These objects provide the primary building blocks for modeling your software assets. A system can encapsulate sub-systems, applications and applications services, databases, components, and forms. An application can encapsulate sub-applications, components, and forms. Databases cannot contain other EAM objects, but can be linked to PDMs, CDMs, and LDMs to model their schemas.



For more information about grouping and organizing EAM objects, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

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, or on the appropriate tab of a system or application. These objects can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- City planning diagram (see [City Planning Diagrams \[page 45\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#)) - as deployment instances (see [Deployment Instances \(EAM\) \[page 80\]](#)).

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:

Table 20:

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

The following tabs are also available:

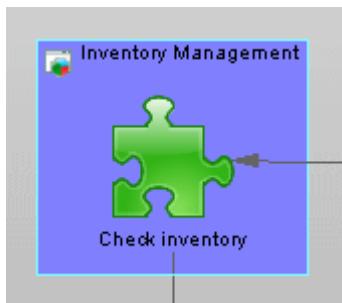
- Detail – [application only] records the programming language, version, etc, for the application.
- Role Associations – lists the people, organization units, and roles associated with the application or system, and the type of role they play in relation to it (see [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).

- 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 contained within the application or system.
- Components – [application and system only] lists the components (see [Components \(EAM\) \[page 61\]](#)) contained within the application or system.
- Forms – [application and system only] lists the forms (see [Forms, Documents, and Reports \(EAM\) \[page 62\]](#)) contained within the application or system.
- Sites – lists the sites associated with the application or system (see [Sites \(EAM\) \[page 30\]](#)).
- Application Services – [system only] lists the application services (see [Application and Business Services \(EAM\) \[page 68\]](#)) contained within the system.
- Databases – [system only] lists the databases contained within the system.
- Documents / Report Documents – [system only] list the documents (see [Forms, Documents, and Reports \(EAM\) \[page 62\]](#)) contained within the system.
- ETL Jobs – [system only] lists the ETL jobs (see [ETL Jobs \(EAM\) \[page 65\]](#)) contained within the system.
- Systems – [system only] lists the sub-systems contained within the system.

4.4 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.

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. Components can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#)) - as deployment instances (see [Deployment Instances \(EAM\) \[page 80\]](#)).

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:

Table 21:

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 <i>Code</i> 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 Lan-guage	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.

4.5 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.

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. These objects can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#)) - as deployment instances (see [Deployment Instances \(EAM\) \[page 80\]](#)).

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:

Table 22:

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 <i>Code</i> 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 [Document Data \(EAM\) \[page 64\]](#)) associated with the document.

4.5.1 Document Data (EAM)

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.

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:

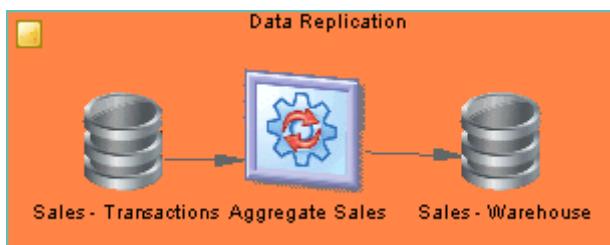
Table 23:

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 <i>Code</i> 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.

4.6 ETL Jobs (EAM)

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

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. ETL jobs can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))

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:

Table 24:

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 <i>Code</i> 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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).

4.7 Contracts (EAM)

A contract is an agreement between services.

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. Contracts can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))

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:

Table 25:

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 <i>Code</i> 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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Documents – lists the documents (see [Forms, Documents, and Reports \(EAM\) \[page 62\]](#)) 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](#).

4.8 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.

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 a business or application service from the Toolbox, Browser, or *Model* menu. Services can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- City planning diagram (see [City Planning Diagrams \[page 45\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))

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:

Table 26:

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

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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Operations – lists the operations (see [Business and Application Service Operations \[page 70\]](#)) that support the service.

4.8.1 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.

Creating an 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.

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:

Table 27:

Property	Description
Parent	Specifies the service to which the operation belongs.
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 <i>Code</i> 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 Forms, Documents, and Reports (EAM) [page 62])

Property	Description
Output	Specifies the document or data returned by the service, which can be linked to an XML model to define its format (see Forms, Documents, and Reports (EAM) [page 62])
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

4.9 Application Links (EAM)

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

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. Application links can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))

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:

Table 28:

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 <i>Code</i> 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.
First object / Second object	Specify the objects at the origin and end of the link.
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

4.10 Importing Application Architecture Metadata from Excel

You can import your systems, applications, and databases from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA_Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

1. To create a new EAM to import into, select . Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting *Import Excel File*. If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the *Select File* tool, browse to and select the Excel file to import, and click *Open* to return to the wizard.

The example file `03 Application Architecture Import.xlsx` contains three sheets:

- **System** - Imports a set of systems that are used to encapsulate applications and databases.
- **EnterpriseApplication** - Imports applications belonging to a system and with values for type, version, and manufacturer.
- **Database** - Imports databases belonging to a system and with values for type, DBMS, and DBMS version.

Note

As all the necessary metadata is contained within the file, selecting the *Auto-map columns to properties* allows the wizard to import all the objects without further intervention.

3. Click the *Options* button set the import options as follows, and click **OK** to return to the wizard:

- *Auto-map columns to properties* (selected)
- All other options - deselected.
- *Reference associated object by: Code*
- *Qualified name separator: .* (dot)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

4. Click *Next* to begin the import.

Progress is displayed in the *Output* window. When the import is complete, a dialog will appear showing how many objects have been created. Click **OK** to return to your model and review the imported objects.

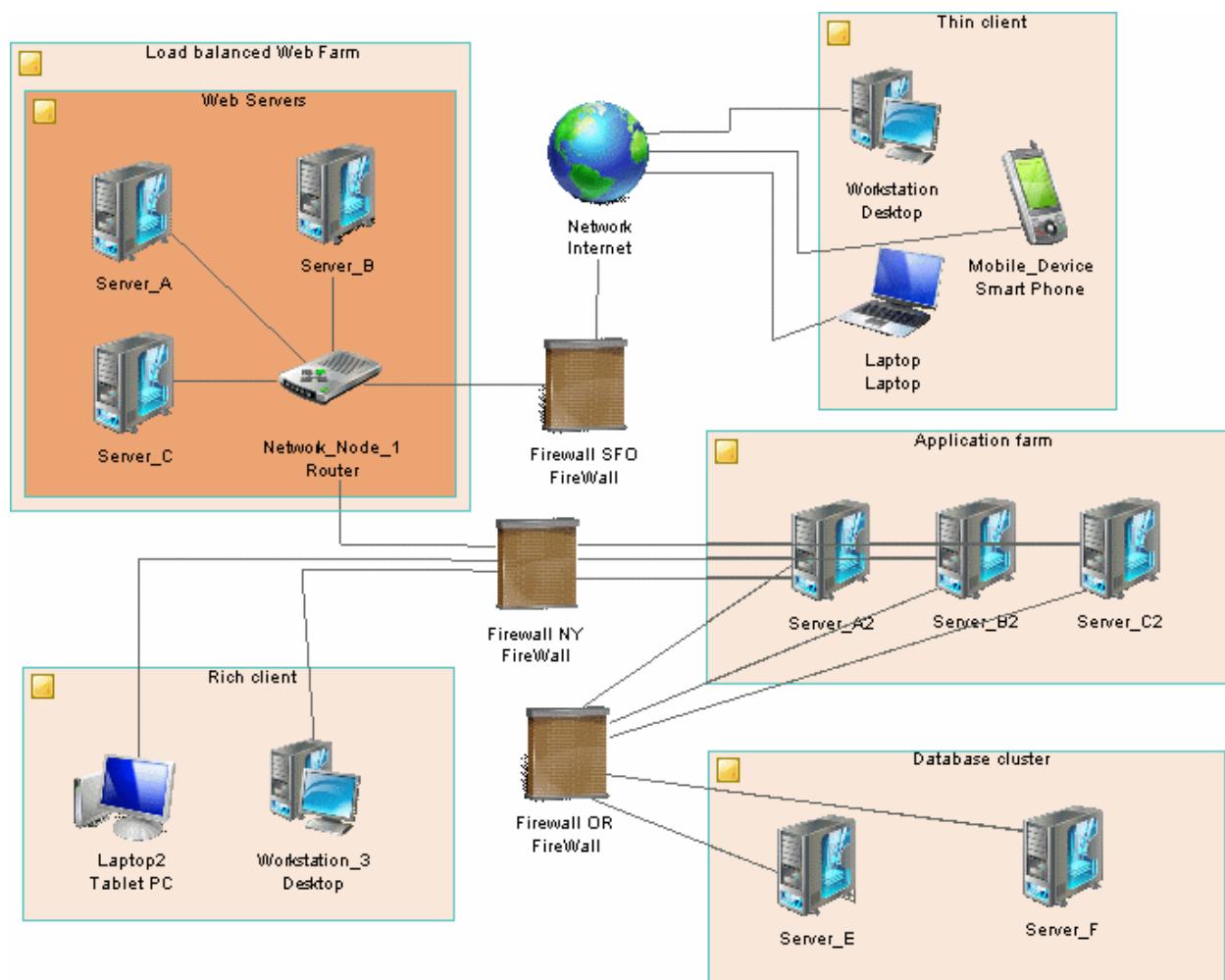
5 Infrastructure Architecture Modeling

PowerDesigner lets you capture, analyze, visualize, and plan changes to the physical infrastructure that supports your activities. You can create technology infrastructure diagrams to provide a big picture view of your networks, servers, firewalls and workstations, the applications and systems that are deployed to them and the connections between them.

5.1 Technology Infrastructure Diagrams

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

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



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**.

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

Table 29:

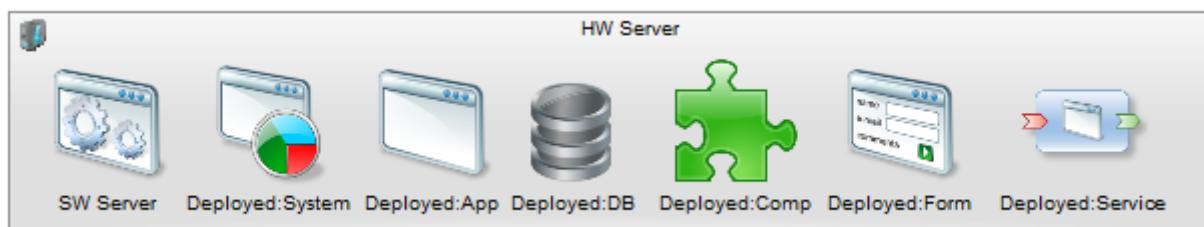
Tool	Description
	<i>Architecture Area</i> - An abstract object for grouping other objects. See Architecture Areas (EAM) [page 46] .
	<i>Site</i> - A physical location. See Sites (EAM) [page 30] .
	<i>Organization Unit</i> - A group, department, function, or other collection of people or organization units. See Organization Units (EAM) [page 32] .
	<i>Person</i> - An individual. See People (EAM) [page 33] .
	<i>Role</i> - A set of responsibilities. See Roles (EAM) [page 39] .
	<i>Network</i> - A LAN, WAN, or other kind of network. See Networks (EAM) [page 78] .
	<i>Software Server</i> - A commercial software environment. See Servers, Workstations, Mobile Devices, and Network Nodes (EAM) [page 76] .
	<i>Hardware Server</i> - A high availability resource machine. See Servers, Workstations, Mobile Devices, and Network Nodes (EAM) [page 76] .
	<i>Workstation</i> - A client machine. See Servers, Workstations, Mobile Devices, and Network Nodes (EAM) [page 76] .
	<i>Mobile Device</i> - A handheld client. See Servers, Workstations, Mobile Devices, and Network Nodes (EAM) [page 76] .
	<i>Network Node</i> - A hardware component connected to a network. See Servers, Workstations, Mobile Devices, and Network Nodes (EAM) [page 76] .
	<i>Deployment Instance</i> - An instance of an object defined elsewhere. See Deployment Instances (EAM) [page 80] .
	<i>Infrastructure Link</i> - A link between elements in a technology layer diagram. See Infrastructure Links (EAM) [page 83] .
	<i>Program</i> - A high-level EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Project</i> - An EA initiative. See Programs, Projects, and Phases (EAM) [page 91] .
	<i>Impact</i> - A link from an EA initiative to the asset that it impacts. See Specifying the Impact of a Project on EA Assets [page 93] .

For more information about objects and diagrams, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

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

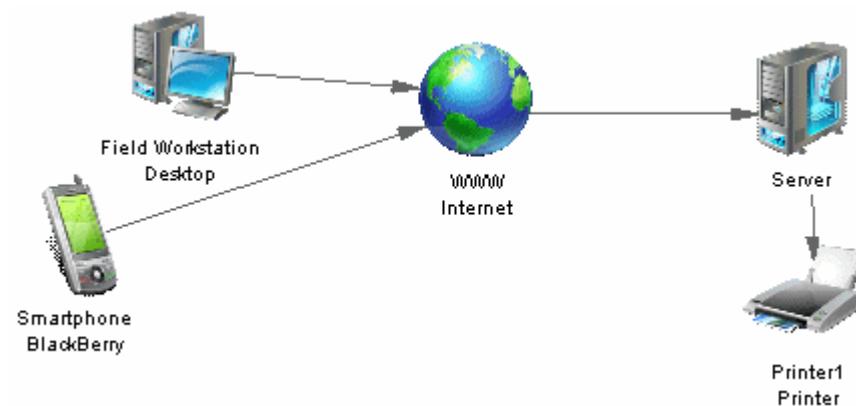
A software server is a commercial software environment, which contains a set of applications and which can be deployed to a hardware server. A hardware server is a device that serves data or other support to network devices and which usually has higher specifications than client devices. A workstation 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).

Servers, workstations, and mobile devices can contain software servers and deployed software assets:



For more information about grouping and organizing EAM objects, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

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:



You can create a server, workstation, mobile device or network node from the Toolbox, Browser, or *Model* menu. These objects can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 30:

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 <i>Code</i> 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 Modeling Cluster Servers and Virtual Machines [page 78] .
Keywords	Provide a way of loosely grouping objects through tagging. To enter multiple keywords, separate them with commas.

The following tabs are also available:

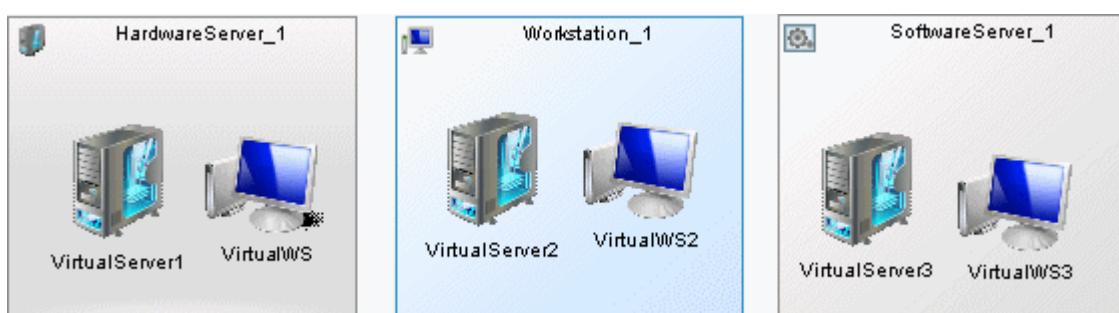
- Detail – contains information to identify the object in the environment.
- Deployment Instances – [not network node] lists the deployment instances (see [Deployment Instances \(EAM\) \[page 80\]](#)) associated with the object.
- Software Servers – [workstation, hardware server, and mobile device only] lists the software servers associated with the object.
- Role Associations – lists the people, organization units, and roles associated with the object (see [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).

5.2.1 Modeling Cluster Servers and Virtual Machines

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

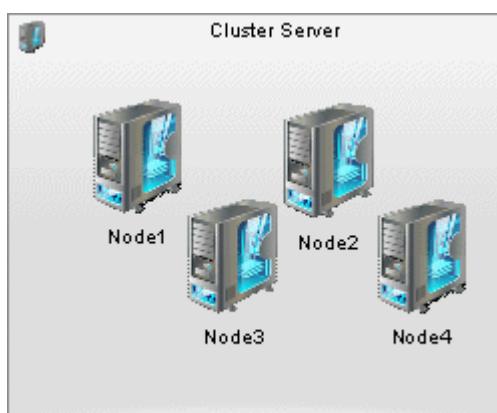
To model virtual machines, create or drag a hardware server or workstation onto a hardware or software server or a workstation. PowerDesigner will select and render read-only the *Virtual* property for objects modeled in this way. You can also model virtual machines in isolation by simply selecting the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

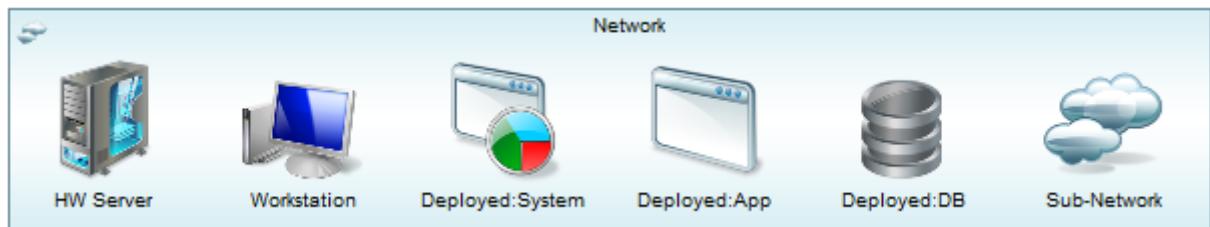
In the following example, the cluster server contains four hardware server nodes:



5.3 Networks (EAM)

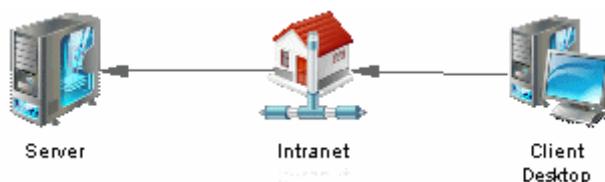
A network allows computers to communicate with each other either locally or over large distances via telecommunications.

Networks can group hardware and deployed software assets:



For more information about grouping and organizing EAM objects, see [Displaying EAM Objects in Diagrams \[page 15\]](#).

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



You can create a network from the Toolbox, Browser, or *Model* menu. Networks can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 31:

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 <i>Code</i> 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.
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 [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- [Deployment Instances](#) – lists the deployment instances associated with the network (see [Deployment Instances \(EAM\) \[page 80\]](#)).
- [Hardware Servers](#), [Software Servers](#), and [Workstations](#) - list the servers and workstations associated with the network (see [Servers, Workstations, Mobile Devices, and Network Nodes \(EAM\) \[page 76\]](#)).

5.4 Deployment Instances (EAM)

Deployment instances are used primarily in technology infrastructure diagrams to represent the deployment of software objects to physical systems. You can deploy a system, application, application service, database, component, or form to multiple servers, workstations, or mobile devices. If you do not want to specify physical systems, you can deploy your software to a software server or architecture area, or simply to the diagram background.

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



Creating a Deployment Instance

To create a deployment instance of a specific software object:

- Drag a system, application, application service, database, component or form from the Browser onto a hardware or software server, workstation, mobile device, or architecture area in a technology infrastructure diagram, or onto the diagram background.
- Open the property sheet of a hardware or software server, workstation, or mobile device, click its *Deployment Instances* tab, and use the *Add Objects* tool.

Once created, deployment instances can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

i Note

You can, alternatively, create a deployment instance without immediately attaching it to a specific deployed object from the Toolbox, Browser, or *Model* menu, or on the *Deployment Instances* tab of a hardware or software server, workstation, or mobile device. You can, subsequently, specify the software object that the deployment instance is deploying by opening the deployment instance property sheet and selecting the object in the *Deployed object* field.

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:

Table 32:

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 <i>Code</i> 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 system, application, application service, database, component or form 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.

5.4.1 Importing Deployment Instances from Excel

You can import deployment instances, which are used to deploy software assets to infrastructure assets in your models, from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA_Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

1. To create a new EAM to import into, select **File** **Import** **Excel File**. Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting *Import Excel File*. If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the **Select File** tool, browse to and select the Excel file to import, and click **Open** to return to the wizard.

The example file 06_Deployment_Instances_Import.xlsx contains two sheets, deploying applications and databases to hardware servers.

Note

Certain metadata in the file cannot be automatically evaluated by the wizard. This file cannot be imported using the *Auto-map columns to properties* option and you must specify the mappings in the wizard.

3. Click the **Options** button set the import options as follows, and click **OK** to return to the wizard:

- All options - deselected.
- Reference associated object by: Code*
- Qualified name separator: .* (dot)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

4. Click **Next** to begin the import.

Specify how your sheets and columns will be imported in the wizard as follows:

- Application Deployments** - Creates deployment instances, linking an application to a hardware server:
 1. Import the table as **Deployment Instance**.

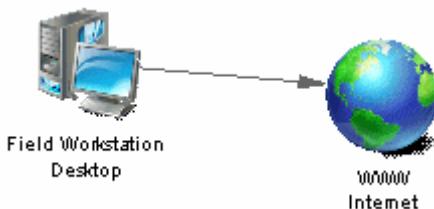
2. Accept the defaults for importing the columns **Parent**, **Name**, and **Code**.
3. Import the column **DeployedObject** as **Deployed Object** and set the *Object* type to **Application**.
 - **Database Deployments** - Creates deployment instances, linking a database to a hardware server:
 1. Import the table as **Deployment Instance**.
 2. Accept the defaults for importing the columns **Parent**, **Name**, and **Code**.
 3. Import the column **DeployedObject** as **Deployed Object** and set the *Object* type to **Database**.

Progress is displayed in the *Output* window. When the import is complete, a dialog will appear showing how many objects have been created. Click **OK** to return to your model and review the imported objects.

5.5 Infrastructure Links (EAM)

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

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. Infrastructure links can be displayed in the following diagrams:

- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))

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:

Table 33:

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 <i>Code</i> 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 /second object	Specify the objects at the origin and 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.

5.6 Importing Infrastructure Architecture Metadata from Excel

An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

- To create a new EAM to import into, select **File** **Import** **Excel File**. Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

i Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting *Import Excel File*. If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the *Select File* tool, browse to and select the Excel file to import, and click *Open* to return to the wizard.

The example file 04 Infrastructure Architecture Import.xlsx contains one sheet:

- o **HardwareServer** - Imports a set of servers deployed to sites and with values for operating system and version, memory, and mass storage.

i Note

As all the necessary metadata is contained within the file, selecting the *Auto-map columns to properties* allows the wizard to import all the objects without further intervention.

3. Click the *Options* button set the import options as follows, and click *OK* to return to the wizard:

- o *Auto-map columns to properties* (selected)
- o All other options - deselected.
- o *Reference associated object by: Code*
- o *Qualified name separator: .* (dot)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

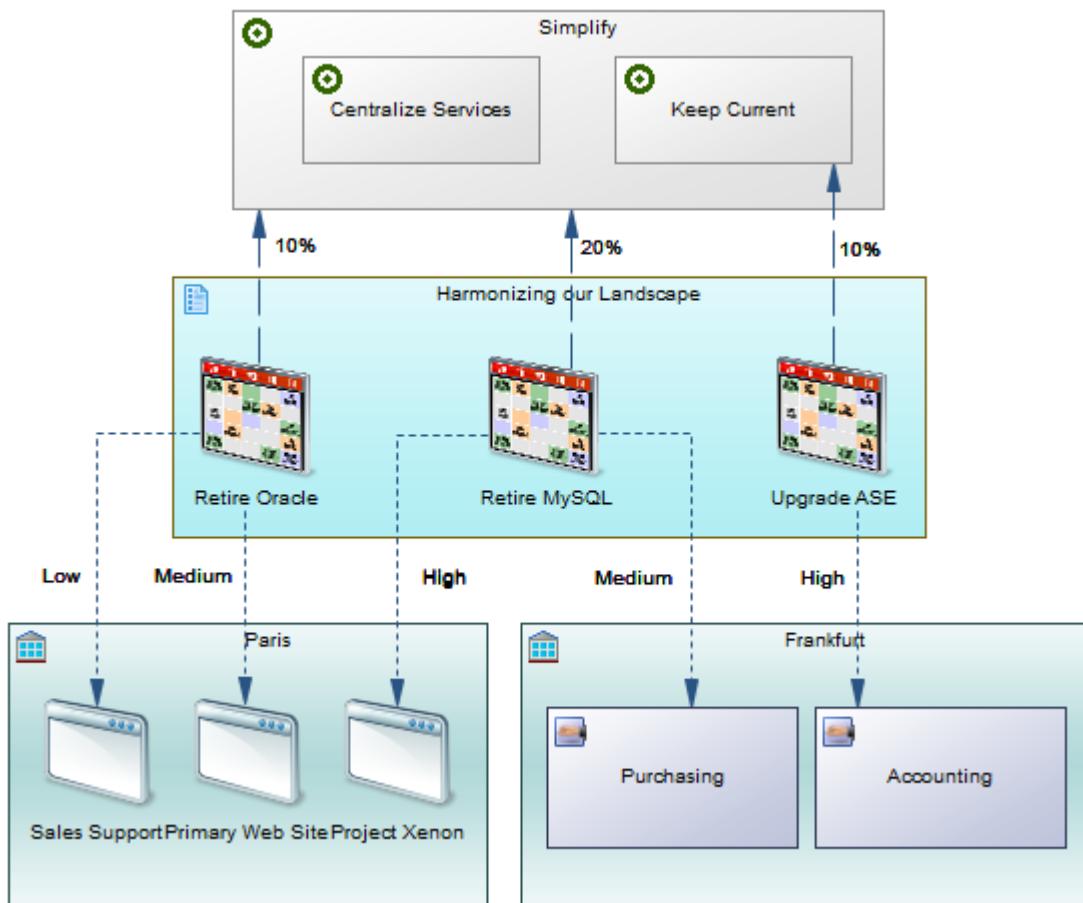
4. Click *Next* to begin the import.

Progress is displayed in the *Output* window. When the import is complete, a dialog will appear showing how many objects have been created. Click *OK* to return to your model and review the imported objects.

6 Goal and EA Project Modeling

PowerDesigner lets you visualize your organization's goals, how your EA projects will help fulfill them, and the impacts those projects will have on your enterprise assets.

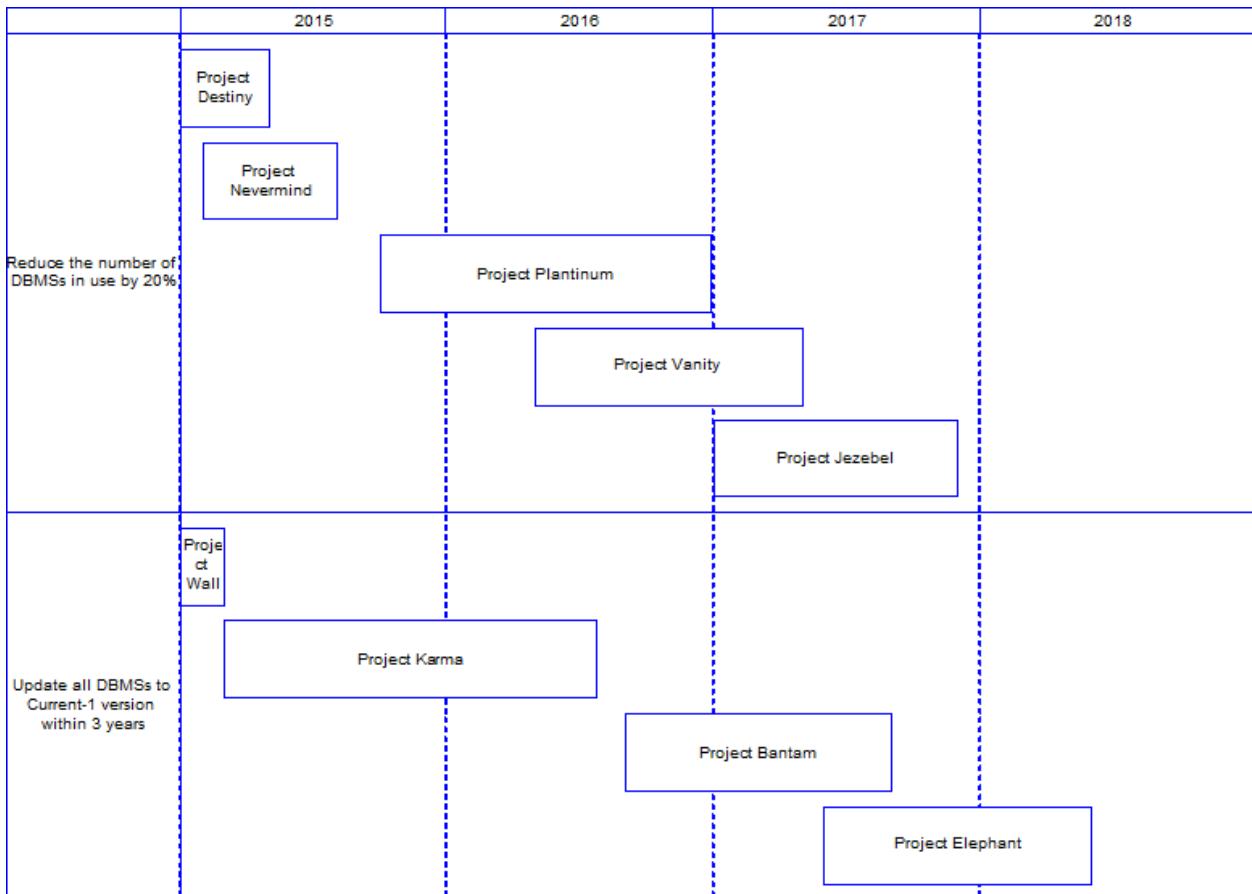
In this example, the Simplify goal contains two sub-goals and we show how the three projects within the Harmonizing our Landscape program help to realize these goals and the impacts that they will have on a number of applications and functions:



6.1 Timeline Diagrams

A *timeline diagram* provides a calendar-style overview of your enterprise architecture programs and projects in relation to your goals, or to the assets that they impact or are in other ways related to.

In the following example, the various projects intended to fulfill two selected goals are displayed in the timeline:



To create a timeline diagram in an existing EAM, right-click the model in the Browser and select **New > Timeline Diagram**, and then complete the following properties:

Table 34:

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.
Period	Specifies the start and end dates that define the limits of the timeline, along with the time units that can be displayed in the header.

Property	Description
Initiative type	<p>Specifies whether projects or programs are displayed on the timeline. If you want to show projects or programs that are not available directly under the model (for example, projects contained in programs) or if you want to specify grouping by another object, you must click the <i>Advanced</i> button and specify the path by which to obtain the projects starting from the model root and passing, by any grouping object.</p> <p>In the example above, goals are modeled in three levels, with third-level goals linked to projects. The type is set to Project [path: Goals (Goal) > SubGoals (Goal) > SubGoals (Goal) > AssociatedProjects (Project)]</p>
Group by	<p>Specifies the objects that are displayed in the left column of the timeline and provide lanes in which the projects or programs are displayed. Click the <i>Selection</i> button to limit the number of lanes displayed.</p> <p>In the example above, grouping is by goals, and two specific goals are selected.</p>

i Note

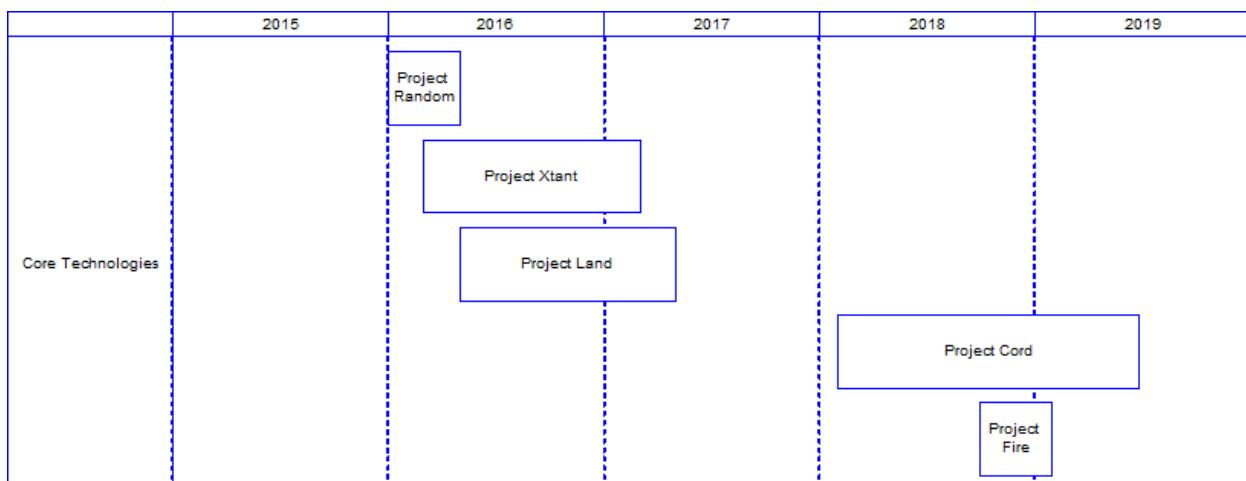
Timeline diagrams are not editable; they are calculated based on the parameters you define in their property sheets. To reopen the property sheet at any time click in the timeline header or in empty space in any of the lanes.

If changes you make to objects displayed in the timeline are not reflected in the timeline, select  **View**  (or press **Shift+F5**) to refresh it.

You can modify the linestyle of the timeline and the look and contents of the project or program symbols by right-clicking the diagram background outside of the timeline and selecting *Display Preferences*.

Example: Projects and Programs

In this example, projects are displayed in relation to the Core Technologies sub-program to which they belong:

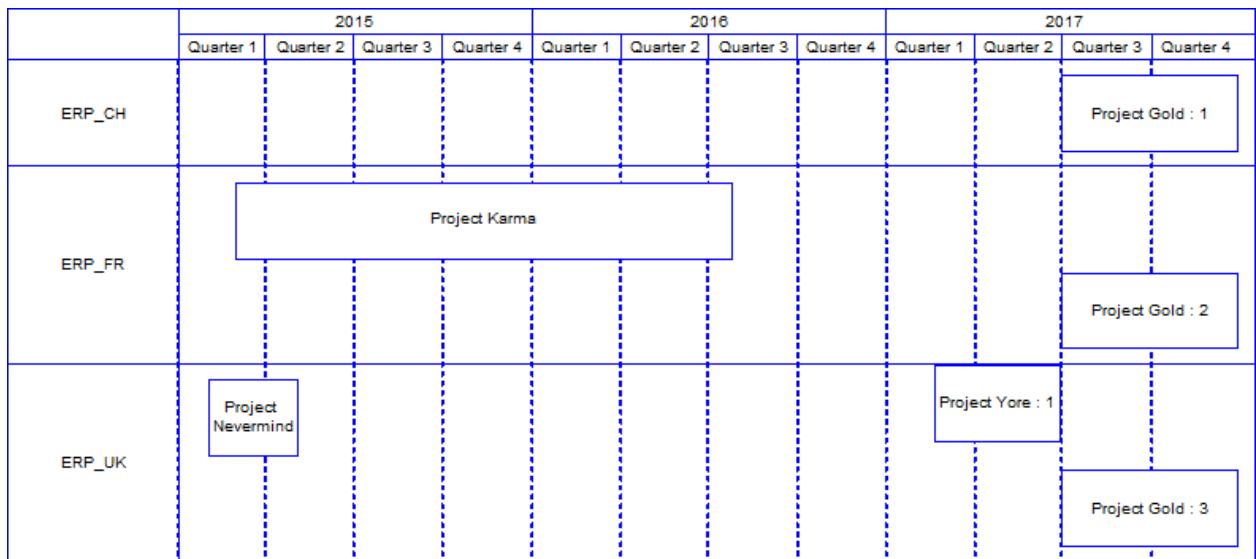


In the model, programs contain sub-programs, which contain projects. The diagram is defined as follows:

- *Initiative type: Project [path: **Programs (Program)** > **SubPrograms (Program)** > **Projects (Project)**]*
- *Group by: **Program** (one program selected)*

Projects and Systems

In this example, projects are displayed in relation to the ERP systems they impact:



In the model, systems are linked to projects by impact links. The diagram is defined as follows:

- *Initiative type: Project [path: **Systems (System)** > **ImpactingProjects (Project)**]*
- *Group by: **System** (three systems selected)*

6.2 Goals (EAM)

Goals allow you to model your organization's mission, vision, strategy, and objectives, to show how they are related to your organization's business and IT architecture, and how they will be addressed through enterprise architecture initiatives.

Creating a Goal

You can create a goal from the Toolbox, Browser, or *Model* menu, or from the *Sub-Goals* tab of a goal. Goals can be displayed in all EAM diagrams:

- Organization chart (see [Organization Charts \[page 27\]](#))
- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))

- Process map (see [Process Maps \[page 43\]](#))
- City planning diagram (see [City Planning Diagrams \[page 45\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))
- Timeline diagram (see [Timeline Diagrams \[page 86\]](#))

Goal Properties

To view or edit a goal'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:

Table 35:

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 <i>Code</i> 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 goal, and the type of role they play in relation to it (see [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Sub-Goals – lists the child goals contained within the goal.
- Projects – lists the contributions of EA programs and projects towards fulfilling the goal (see [Specifying the Fulfillment of Goals by a Project \[page 92\]](#)).

6.3 Programs, Projects, and Phases (EAM)

Programs and projects represent EA initiatives that are intended to fulfill corporate goals and may impact elements of your organization's business and IT architecture. Programs can contain sub-programs and projects, and projects can contain phases.

Creating a Program, Project, or Phase

You can create a program or project from the Toolbox, Browser, or *Model* menu. Projects can be created on the *Projects* tab of a program, and phases on the *Phases* tab of a project. These objects can be displayed in all EAM diagrams:

- Organization chart (see [Organization Charts \[page 27\]](#))
- Business communication diagram (see [Business Communications Diagrams \[page 28\]](#))
- Process map (see [Process Maps \[page 43\]](#))
- City planning diagram (see [City Planning Diagrams \[page 45\]](#))
- Application architecture diagram (see [Application Architecture Diagrams \[page 54\]](#))
- Service-oriented diagram (see [Service-Oriented Diagrams \[page 56\]](#))
- Technology infrastructure diagram (see [Technology Infrastructure Diagrams \[page 74\]](#))
- Timeline diagram (see [Timeline Diagrams \[page 86\]](#))

Program, Process, and Phase Properties

To view or edit these objects' properties, double-click their 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:

Table 36:

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 <i>Code</i> 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.
Start date/End date	Specify the duration of the program, project, or phase.

Property	Description
Progress	Specifies the current advancement of the program, project, or phase.
Status	Specifies the present status of the program, project, or phase.
IT Capital Expenditure/IT Operating Expenditure/Business Expenditure	Specify values for each of these types of expenditure for the program, project, or phase.
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 program, project, or phase, and the type of role they play in relation to it (see [Associating a Person, Organization Unit, or Role with an Object \[page 35\]](#)).
- Sub-Programs - [programs] lists the child programs under the program. Use the [Add a Row](#) tool on this tab to create a new program.
- Projects - [programs] lists the projects under the program. Use the [Add a Row](#) tool on this tab to create a new project.
- Impacts - lists the EA assets that are impacted by the program, project, or phase (see [Specifying the Impact of a Project on EA Assets \[page 93\]](#)).
- Associated Goals - lists the goals that the program, project, or phase contributes to fulfilling (see [Specifying the Fulfillment of Goals by a Project \[page 92\]](#)).

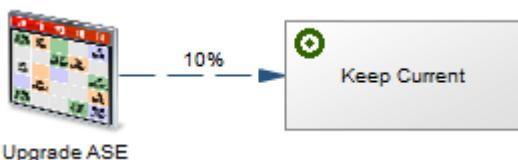
6.3.1 Specifying the Fulfillment of Goals by a Project

You can specify the fulfillment of goals by projects (or programs or phases) using the Toolbox [Fulfillment](#) tool or on the [Goals](#) tab of the property sheet of the project. The fulfillment link is displayed in the diagram, and the list of projects contributing to the fulfillment of a goal is available from the [Dependencies](#) tab of the property sheet of the goal.

Context

To create the fulfillment directly in a business communication diagram or process map, click the [Fulfillment](#) tool in the Toolbox, and draw a link from the project (or program or phase) to the goal. Double-click the link to open its property sheet and specify a fulfillment percentage and description.

In this example the Upgrade ASE project fulfills 10% of the Keep Current goal:



To specify fulfillment of goals in any diagram, including those where the *Fulfillment* tool is not available:

Procedure

1. Open the property sheet of the project (or program or phase) that is contributing to the fulfillment of a goal and select the *Goals* tab.
2. Click the *Add Objects* tool to open a dialog listing the available goals, select the goals that the project will realize, and click *OK* to return to the *Goals* tab.
3. [optional] Specify a fulfillment percentage and description to explain how the project will fulfill the goal.
4. Click *OK* to return to the diagram.

If the project and goal are in the current diagram then the link will be drawn between them.

i Note

To display programs, projects, and goals in a diagram drag them from the Browser, or select ► *Symbol* ► *Show Symbols* ▾ and choose the relevant objects from the dialog. To display fulfillment, impact, and other links in the diagram, select ► *Tools* ► *Complete Links* ▾.

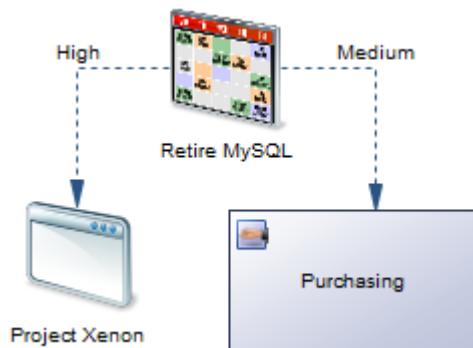
6.3.2 Specifying the Impact of a Project on EA Assets

You can specify the impact of projects (or programs or phases) on your enterprise assets (any object in the EAM) using the Toolbox *Impact* tool or on the *Impacts* tab of the project property sheet. The impact link is displayed in the diagram, and the list of projects impacting an asset is available from the *Dependencies* tab of the property sheet of the asset.

Context

To create the impact directly in the diagram, click the *Impact* tool in the Toolbox, and draw a link from the project (or program or phase) to the asset. Double-click the link to open its property sheet and specify an impact level and description.

In this example the Retire MySQL project has a high impact on the Project Xenon application and a medium impact on the Purchasing business function:



To specify impacts on assets from the property sheet of a project, program, or phase:

Procedure

1. Open the property sheet of the project (or program or phase) that is impacting an EA asset and select the *Impacts* tab.
2. Click the *Add Objects* tool to open a dialog listing the available assets, select those that the project will impact, and click *OK* to return to the *Impacts* tab.
3. [optional] Specify an impact level and description to explain how the project will impact the asset.
4. Click *OK* to return to the diagram.

If the project and asset are in the current diagram then the link will be drawn between them.

Note

To display programs, projects, and goals in a diagram drag them from the Browser, or select ► *Symbol* ► *Show Symbols* ▾ and choose the relevant objects from the dialog. To display fulfillment, impact, and other links in the diagram, select ► *Tools* ► *Complete Links* ▾.

6.4 Importing Corporate Goals and EA Projects from Excel

You can import goals and the programs and projects that will seek to fulfill them from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA_Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

1. To create a new EAM to import into, select Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting *Import Excel File*. If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the *Select File* tool, browse to and select the Excel file to import, and click *Open* to return to the wizard.

The example file `07 Goals and Projects Import.xlsx` contains three sheets:

- **Goal** - Imports a three-level set of corporate goals.
- **Program** - Imports a two-level set of programs that will contain projects for fulfilling the goals.
- **Project** - Imports a set of projects that belong to second-level programs.

Note

As all the necessary metadata is contained within the file, selecting the *Auto-map columns to properties* allows the wizard to import all the objects without further intervention.

3. Click the *Options* button set the import options as follows, and click **OK** to return to the wizard:

- *Auto-map columns to properties* (selected)
- All other options - deselected.
- *Reference associated object by: Code*
- *Qualified name separator: .* (dot)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

4. Click *Next* to begin the import.

Progress is displayed in the *Output* window. When the import is complete, a dialog will appear showing how many objects have been created. Click **OK** to return to your model and review the imported objects.

6.4.1 Importing Impacts, Fulfillments, and Stakeholders from Excel

You can import links from projects to other objects in your models from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

- To create a new EAM to import into, select Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting *Import Excel File*. If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

- Click the *Select File* tool, browse to and select the Excel file to import, and click *Open* to return to the wizard.

The example file 08 Impacts Fulfillments and Stakeholders Import.xlsx contains four sheets linking programs and projects to various other objects created in precedent imports.

Note

Certain metadata in the file cannot be automatically evaluated by the wizard. This file cannot be imported using the *Auto-map columns to properties* option and you must specify the mappings in the wizard.

- Click the *Options* button set the import options as follows, and click *OK* to return to the wizard:

- All options - deselected.
- Reference associated object by: Code*
- Qualified name separator: . (dot)*

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

- Click *Next* to begin the import.

Specify how your sheets and columns will be imported in the wizard as follows:

- Program.Sponsors** - Specifies people as sponsors of programs:
 - Import the table as **Program.Role Association**.
 - Import the column **Parent** as **Parent**.
 - Import the column **Role Player** as **Role Player** and set the *Object* type to **Person**.
 - Import the column **Type** as **Type**.
- Project.Sponsors** - Specifies people as sponsors of projects:
 - Import the table as **Project.Role Association**.
 - Import the column **Parent** as **Parent**.
 - Import the column **Role Player** as **Role Player** and set the *Object* type to **Person**.
 - Import the column **Type** as **Type**.
- Project.Impact** - Links projects to the systems that they will impact:
 - Import the table as **Project.Impact**.
 - Import the column **Parent** as **Parent**.
 - Import the column **EnterpriseAsset** as **Enterprise Asset** and set the *Object* type to **System**.
 - Import the column **ImpactLevel** as **Impact Level**.
- Project.Fulfillment** - Links projects to the goals that they will fulfill:
 - Import the table as **Project.Fulfillment**.

2. Import the column **Parent** as **Parent**.
3. Import the column **Goal** as **Goal**.
4. Import the column **Fulfillment** as **Fulfillment**.

Progress is displayed in the *Output* window. When the import is complete, a dialog will appear showing how many objects have been created. Click **OK** to return to your model and review the imported objects.

7 SAP IQ Reference Architecture Model

PowerDesigner provides a special EAM model to help you analyze the architecture required to deploy a SAP® 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.

7.1 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.

Procedure

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

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

7.2 Creating Reference Architectures

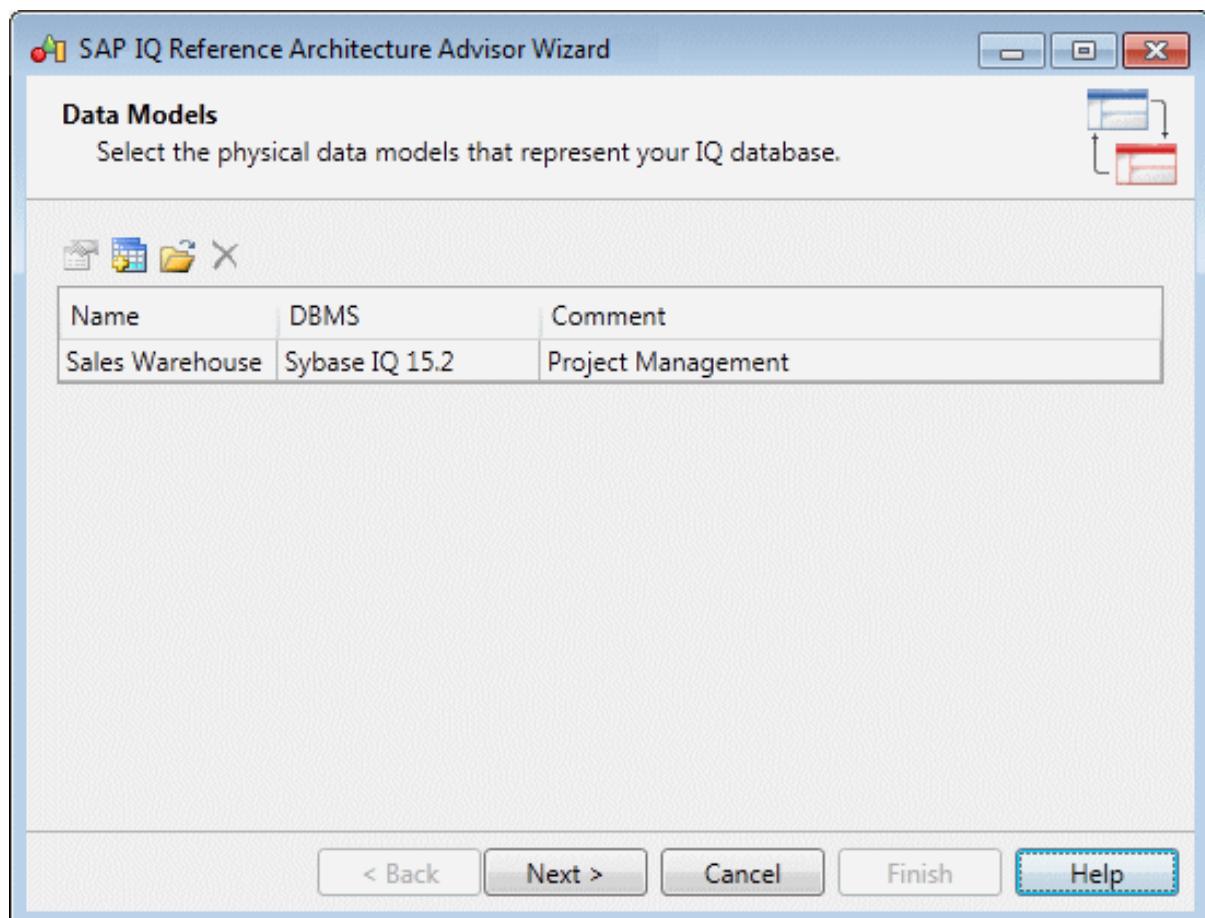
PowerDesigner provides a wizard to let you generate one or more reference architectures appropriate for your IQ environment, each based on a different hardware profile. The resulting diagram and objects detail the costs and specifications for all the elements required for each reference architecture.

Context

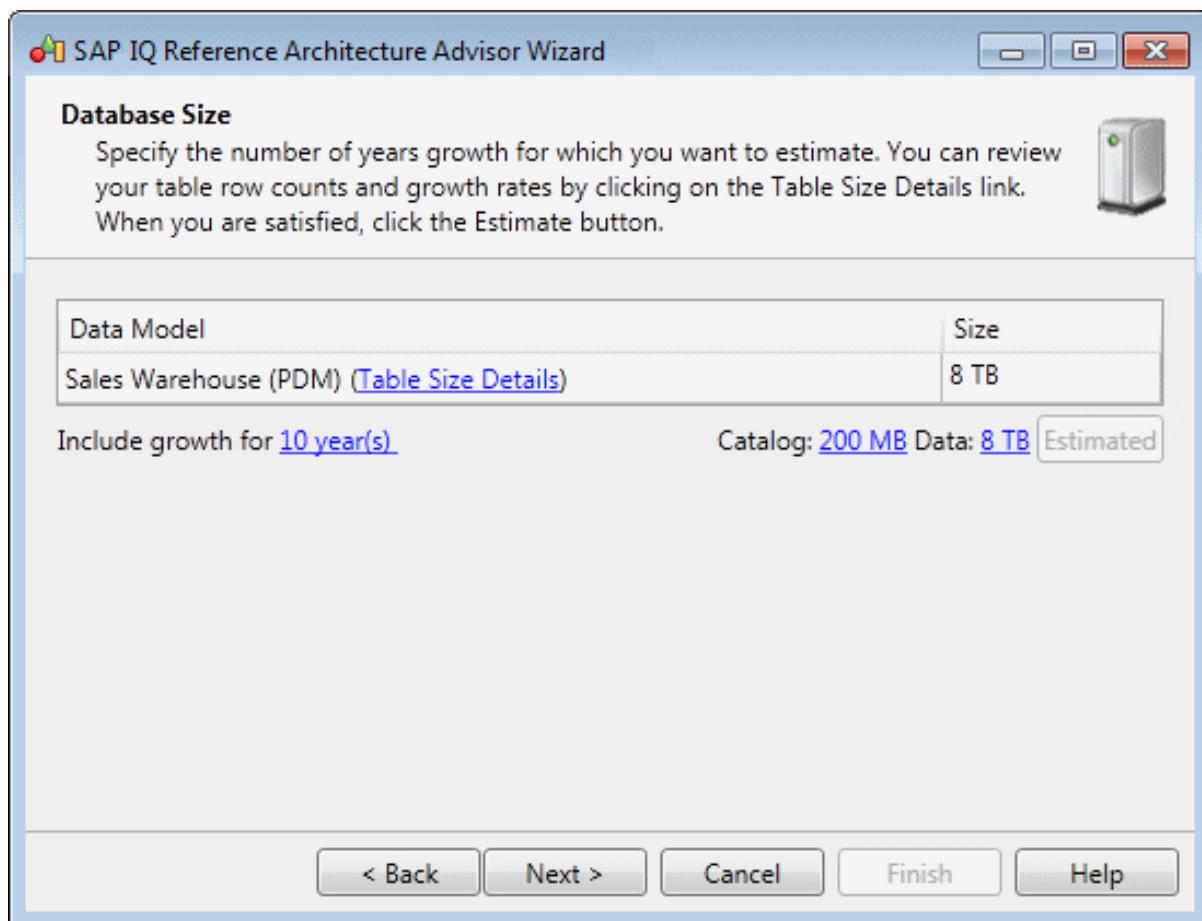
Before launching the advisor wizard, you must develop a physical data model to represent your IQ database and have specified appropriate candidate machines in the hardware profile library (see [Hardware Profile Library \[page 113\]](#)).

Procedure

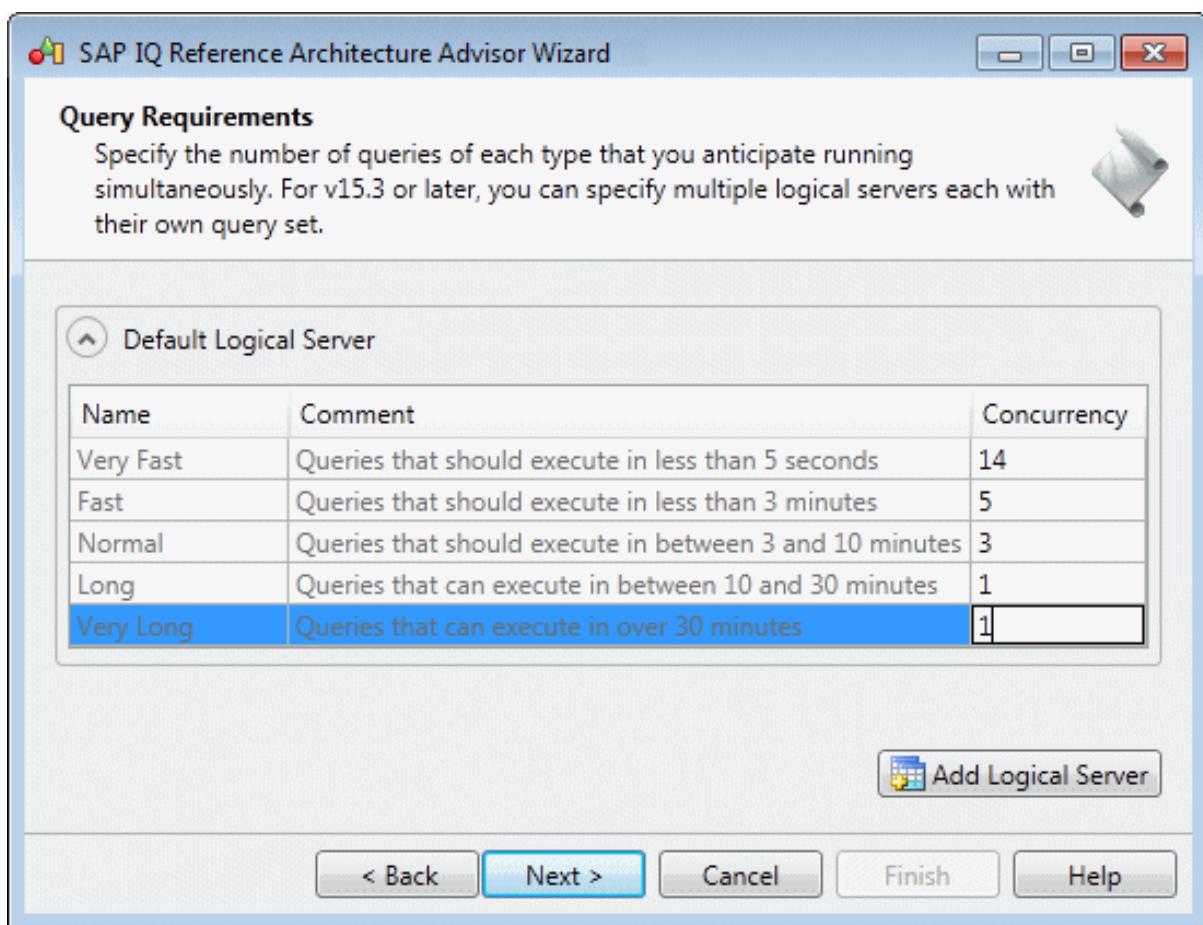
1. Select **Tools > SAP IQ Reference Architecture > Launch Advisor Wizard** to open the wizard.
2. Click **Next** to go to the **Data Models** page, and use the **Select Data Models** tool to add a PDM from the workspace or the **Select Data Models from Files** tool to open a PDM:



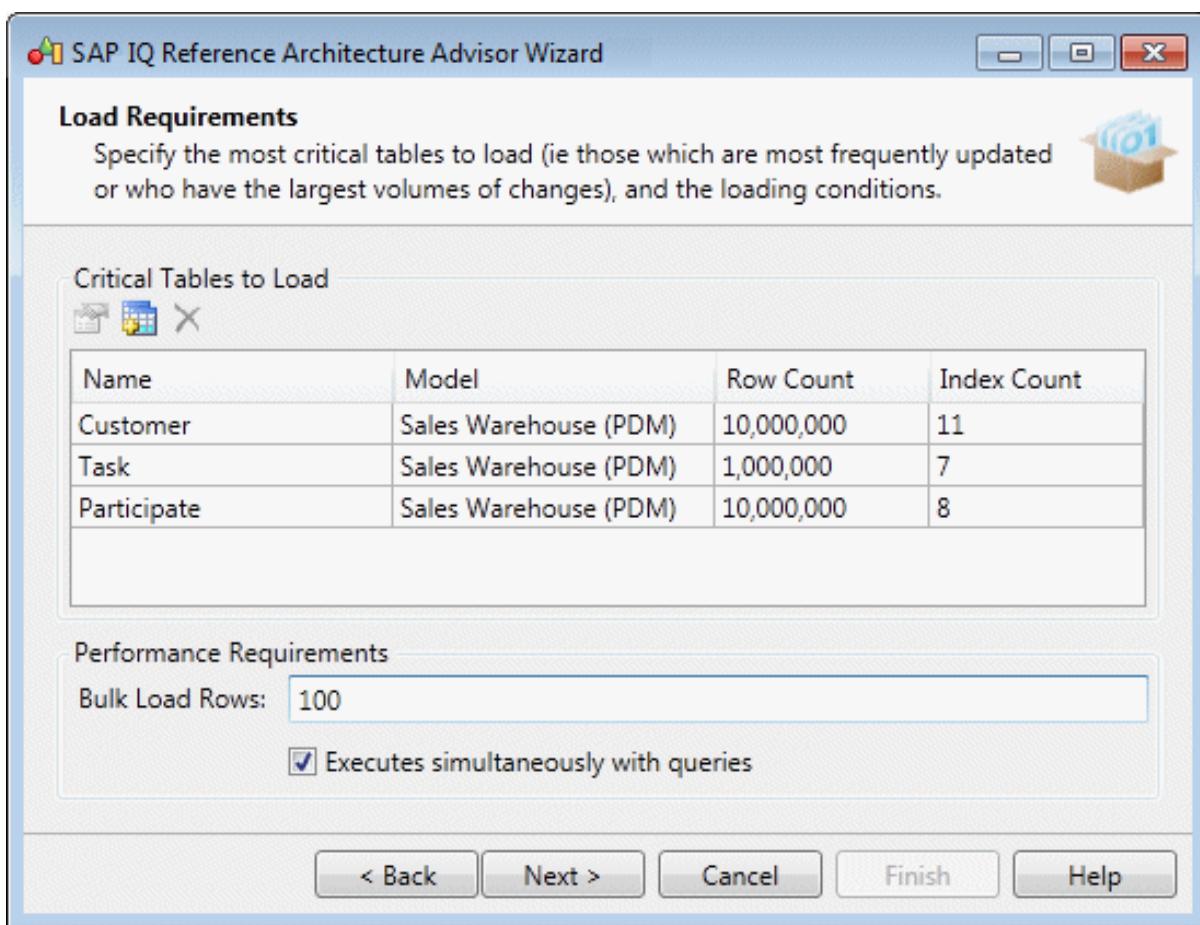
- Click **Next** to go to the **Database Size** page, and specify the number of years growth for which you want to estimate. To review and edit table row counts and growth rates, click the **Table Size Details** link. When you are satisfied with your figures, click the **Estimate** button to produce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



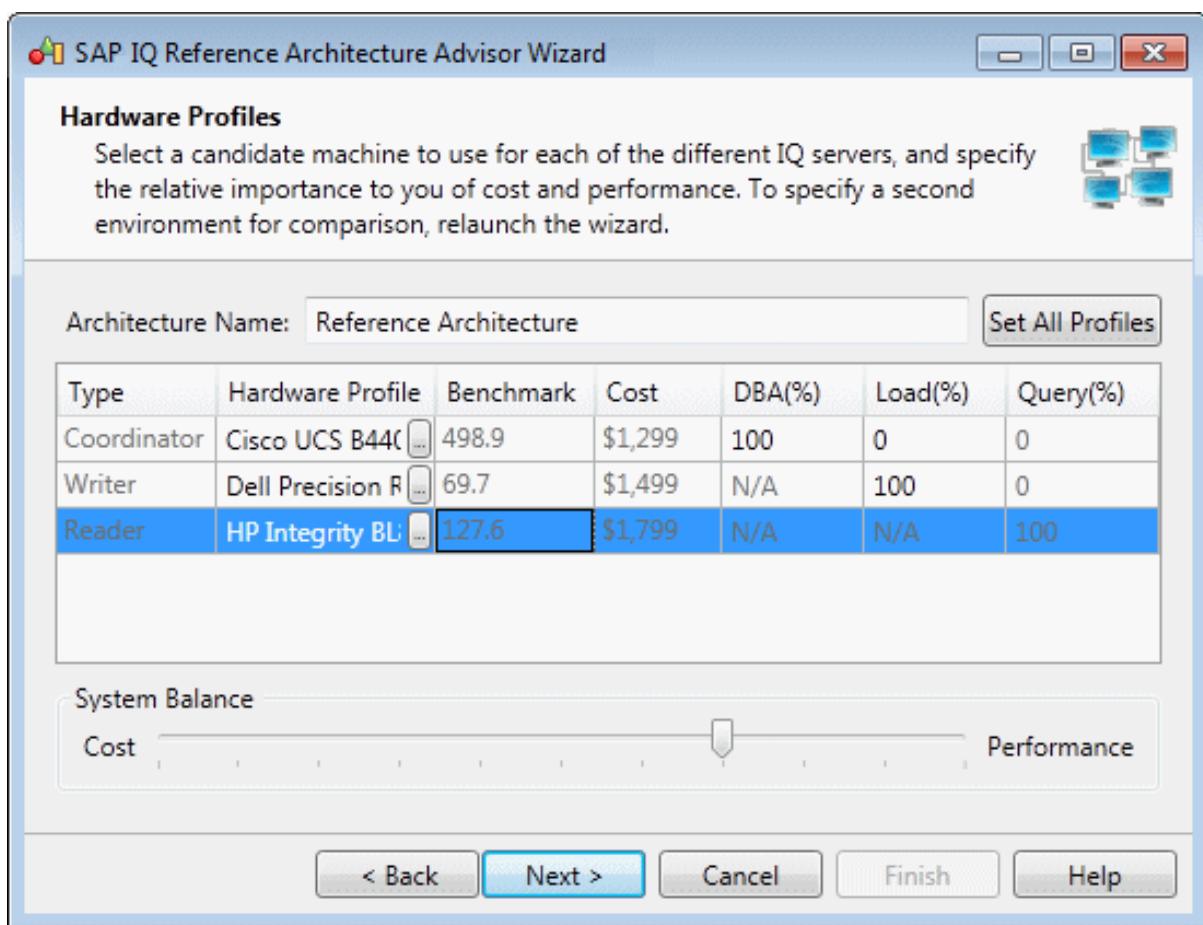
- Click **Next** to go to the **Query Requirements** page, and specify the number of queries of each type that you anticipate running simultaneously:



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:



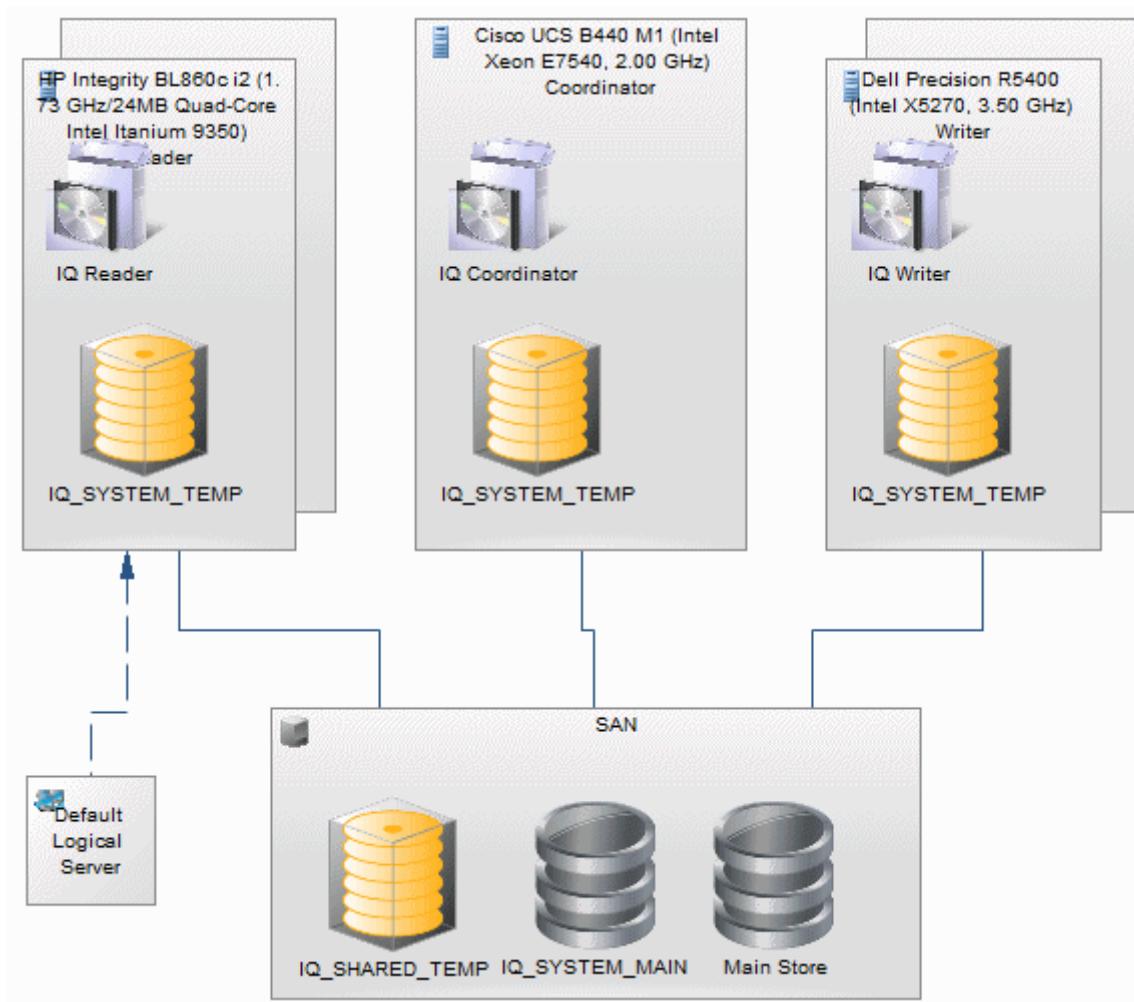
8. Click *Next* to go to the *Summary* page, which lists the information entered on the preceding pages.

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:



i Note

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

7.2.1 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.

i 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 \[page 112\]](#)).

The *General* tab contains the following properties:

Table 37:

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 <i>Code</i> 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.

7.2.2 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:

Table 38:

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 <i>Code</i> field.
Hardware profile	[read-only] Specifies the hardware profile (see Hardware Profile Properties [page 114]), 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 \[page 106\]](#)).
- *IQ Stores* – Lists the IQ stores defined on the server (see [IQ Store Properties \[page 107\]](#)).

7.2.3 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:

Table 39:

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 <i>Code</i> 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:

Table 40:

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.
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.

7.2.4 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:

Table 41:

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 <i>Code</i> field.
Cost	[read-only] Specifies the total cost of the SAN, based on the price specified on the model property sheet (see Reference Architecture Model Properties [page 108]).

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 \[page 107\]](#)).

7.2.5 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:

Table 42:

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 <i>Code</i> 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.

7.2.6 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:

Table 43:

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 <i>Code</i> field.
Concurrency	Specifies the number of this type of query that the environment must be able to perform simultaneously.

7.2.7 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 \[page 12\]](#)) in addition to the following properties:

Table 44:

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:

Table 45:

Property	Description
Main cache ratio for reader/ writer (%)	Specifies the percentage of the cache allocated to the main (as opposed to temporary) cache.

Property	Description
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_SYSTEM_MAIN (GB)	Specifies the standard and maximum sizes (in GB) allocated to the IQ_SYSTEM_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.

7.2.8 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:

Table 46:

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 <i>Code</i> 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:

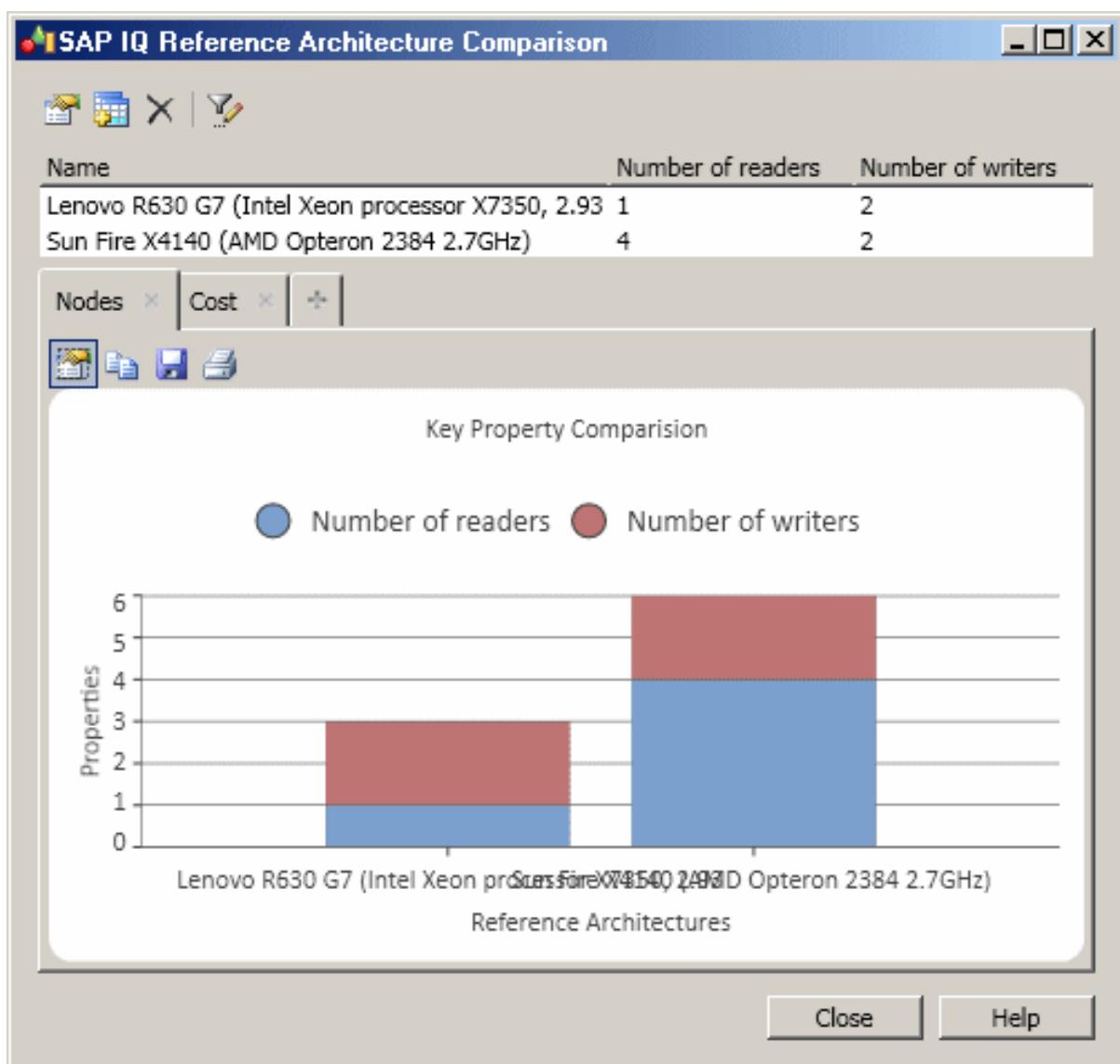
- *Query Requirements* - Lists the number of each type of query that must be supported concurrently.

7.3 Comparing Reference Architectures

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

Procedure

1. Select  *Tools*  *SAP IQ Reference Architecture*  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:



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

7.4 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.

Procedure

1. Modify any of the generated properties on the reference architecture property sheet (see [Reference Architecture Properties \[page 104\]](#)), 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.

i 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 ► **SAP 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.

7.5 Generating Reports from a Reference Architecture Model

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

8 Hardware Profile Library

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.

8.1 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.

Context

i Note

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

Procedure

1. In your Web browser, go to the benchmark search form <http://www.spec.org/cgi-bin/osgrresults?conf=int2006&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.

i 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.

i 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 \[page 99\]](#)).

8.1.1 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:

Table 47:

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 <i>Code</i> 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.

Property	Description
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.

8.2 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 \[page 12\]](#)) in addition to the following properties:

Table 48:

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

SAP IQ Configuration Tab

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

Table 49:

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).

8.2.1 Specifying Additional Query Types

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

Procedure

1. Open the property sheet of the hardware profile library and click the *SAP 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).

The next time that you launch the Advisor wizard in an reference architecture model associated with the hardware library (see [Creating Reference Architectures \[page 99\]](#)), you will be able to specify requirements for the new query type.

9 Checking an EAM

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

i 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*.

9.1 Package Checks

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

Table 50:

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.

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 <i>Result List</i> 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.

9.2 Application, Application Service, Component and Database Checks

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

Table 51:

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 at least one deployment instance.</p> <ul style="list-style-type: none"> Manual correction: Deploy the object as a deployment instance. Automatic correction: None

9.3 Application Link Checks

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

Table 52:

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.

9.4 Contract Checks

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

Table 53:

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.

Check	Description and Correction
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

9.5 Deployment Instance Checks

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

Table 54:

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.

Check	Description and Correction
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 <i>General</i> 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

9.6 Form Checks

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

Table 55:

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

9.7 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.

Table 56:

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.
Serial number uniqueness	Serial numbers must be unique in the model. <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	Asset numbers must be unique in the model. <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	Mac addresses must be unique in the model. <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	The object must be linked with at least one other node. <ul style="list-style-type: none">• Manual correction: Link the object to an appropriate object.• Automatic correction: None

9.8 Network Checks

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

Table 57:

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.
Isolated node	The object must be linked with at least one other node. <ul style="list-style-type: none">• Manual correction: Link the object to an appropriate object.• Automatic correction: None

9.9 Organization Unit Checks

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

Table 58:

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.

Check	Description and Correction
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.

9.10 Person Checks

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

Table 59:

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

9.11 Software Server Checks

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

Table 60:

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.
Software Server not deployed	A software server must be deployed to a physical server. <ul style="list-style-type: none">• Manual correction: Deploy the software server in the appropriate object.• Automatic correction: None
Software Server serial number uniqueness	Software Server serial numbers must be unique in the model. <ul style="list-style-type: none">• Manual correction: Modify the duplicate serial numbers.• Automatic correction: Appends a number to the duplicate serial numbers.

10 Importing and Linking to and Generating and Exporting from an EAM

PowerDesigner provides various tools to import to and generate and export from your EAM, and to link your EA objects to objects in other models.

10.1 Importing EA Metadata from Excel

While you can manually create objects to represent all the sites, servers, and apps and other assets of your enterprise, these kinds of lists or catalogs are commonly collected in spreadsheets and imported en masse.

PowerDesigner is delivered with a set of nine Excel files containing sample data, which can be imported into one or more EAMs:

- 01 Organization Architecture Import.xlsx - To import sites, organization units, and people (see [Importing Organization Architecture Metadata from Excel \[page 41\]](#)).
- 02 Process Architecture Import.xlsx - To import processes, business functions, and architecture areas (see [Importing Process Architecture Metadata from Excel \[page 52\]](#)).
- 03 Application Architecture Import.xlsx - To import systems, applications, and databases (see [Importing Application Architecture Metadata from Excel \[page 72\]](#)).
- 04 Infrastructure Architecture Import.xlsx - To import hardware servers (see [Importing Infrastructure Architecture Metadata from Excel \[page 84\]](#)).
- 05 Role Associations Import.xlsx - To import role associations to define people as owners of processes, functions, servers, applications, and databases. The import of role associations does not support auto-mapping. You must specify the import manually (see [Importing Role Associations from Excel \[page 37\]](#)).
- 06 Deployment Instances Import.xlsx - To import deployments of applications and databases to hardware servers. The import of deployment instances does not support auto-mapping. You must specify the import manually (see [Importing Deployment Instances from Excel \[page 82\]](#)).
- 07 Goals and Projects Import.xlsx - To import corporate goals, and EA programs and projects (see [Importing Corporate Goals and EA Projects from Excel \[page 94\]](#)).
- 08 Impacts Fulfillments and Stakeholders Import.xlsx - To import these links between projects and other EA assets. These imports do not support auto-mapping. You must specify the import manually (see [Importing Impacts, Fulfillments, and Stakeholders from Excel \[page 95\]](#)).
- 09 Traceability Links Import.xlsx - To import traceability links between various EA objects. The import of traceability links does not support auto-mapping. You must specify the import manually (see [Importing Traceability Links from Excel \[page 134\]](#)).

These files are delivered as part of the EA Example files available at `<install_dir>/Examples/EAExample` (see [Example EA Model, Imports, and Charts \[page 19\]](#)).

During the import, in addition to the objects that are created, PowerDesigner creates an Excel Import object that contains your import parameters, along with a Table Mapping object for each of the imported tables, which lists the column mappings used. Right-click the Excel Import Object to access the following commands:

- [Import](#) - to re-import your file based on the options and mappings you have already defined. All the objects will be reimported, overwriting any existing objects based on their names. If you have added new objects to your worksheet tables they will be added to the model, but any objects that you have deleted from your worksheet will not be deleted from the model.
- [Change Mappings](#) - to re-import your file via the Excel Import Wizard in order to modify your table or column mappings.
- [Change Options](#) - to change the import options preparatory to re-importing with the command [Import](#).
- [Properties](#) - to open the Excel Import properties sheet.

You can import any number of different types of objects from a single Excel file, so long as they can all be imported to a single type of model. Each type of object should be listed in its own table on a separate worksheet of the Excel file. Each row in a table represents one object to import, and each column represents one property (an attribute or list of associated objects) of the object.

To reuse any of these files for your own imports, simply delete the sample data and enter your own. You can add further columns in a table to import other properties, and add further worksheets to import other types of objects. For detailed information about working with PowerDesigner and Excel, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

10.2 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.

Context

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:

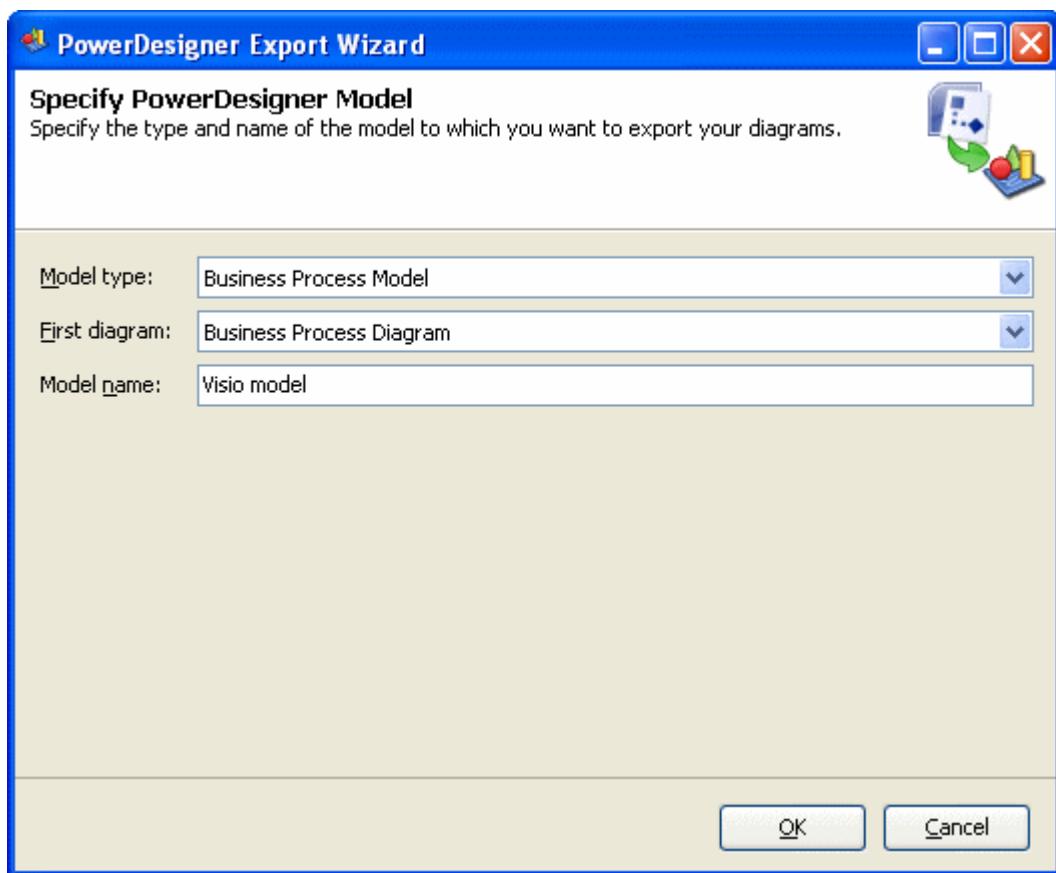
Table 61:

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

Visio Template	PowerDesigner Diagram
Business Process/ Data Flow Diagram	BPM Data Flow Diagram
Software/ 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
LDAP Directory	EAM Organization Chart Diagram

Procedure

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



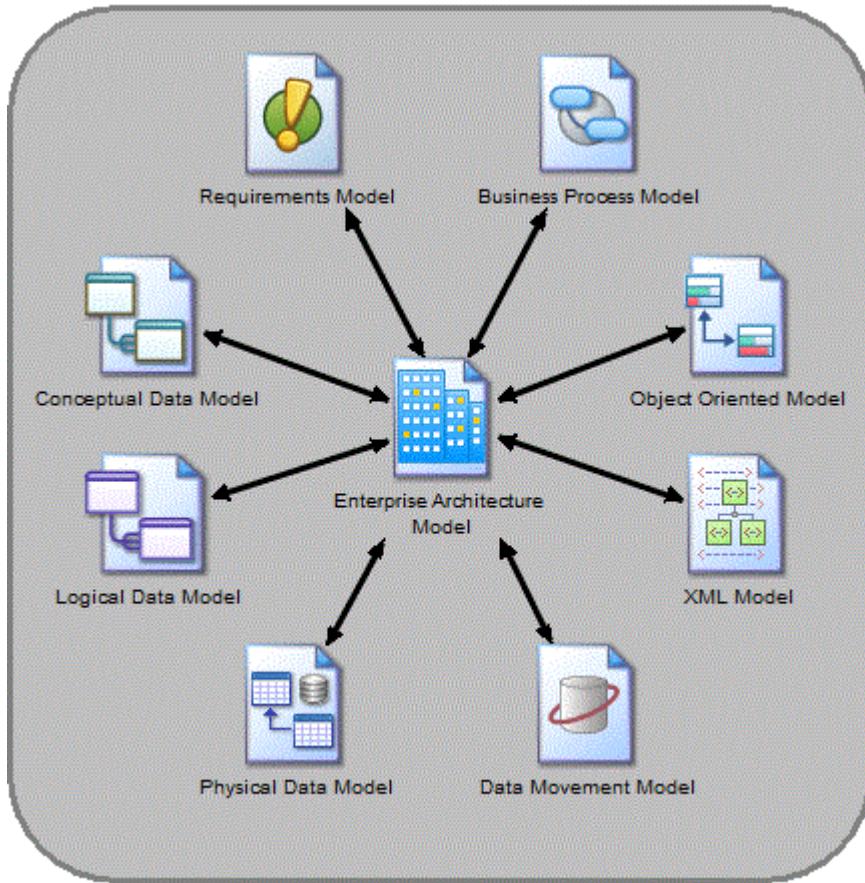
2. Specify the type of model to which you want to export your diagram, enter a name for the model to be created, and then click **OK** to start the export.
3. When the export is complete, click **OK** to close the wizard.

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

10.3 Exporting and Importing Objects to and from Other Models

While the EAM provides a big picture of your organization and permits you to decompose your functions, processes, and systems to a certain extent, it is not suitable for showing detailed technical implementation information. When modeling the implementation of databases, processes, web services, and other assets, you will use the appropriate specialized PowerDesigner model.

The EAM provides wizards to export EA objects to these other, lower-level models and to import implementation objects into the EAM. Objects exported or imported in this way retain links to the original objects 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:

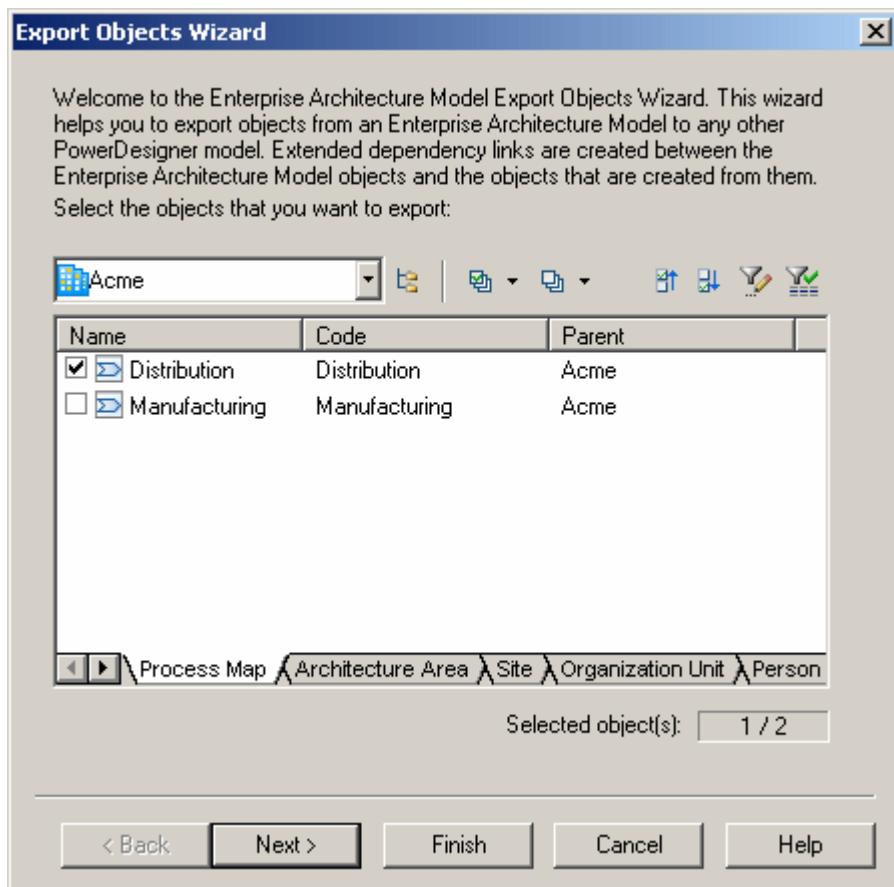


10.3.1 Exporting Objects from an EAM to Another Model

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

Procedure

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.

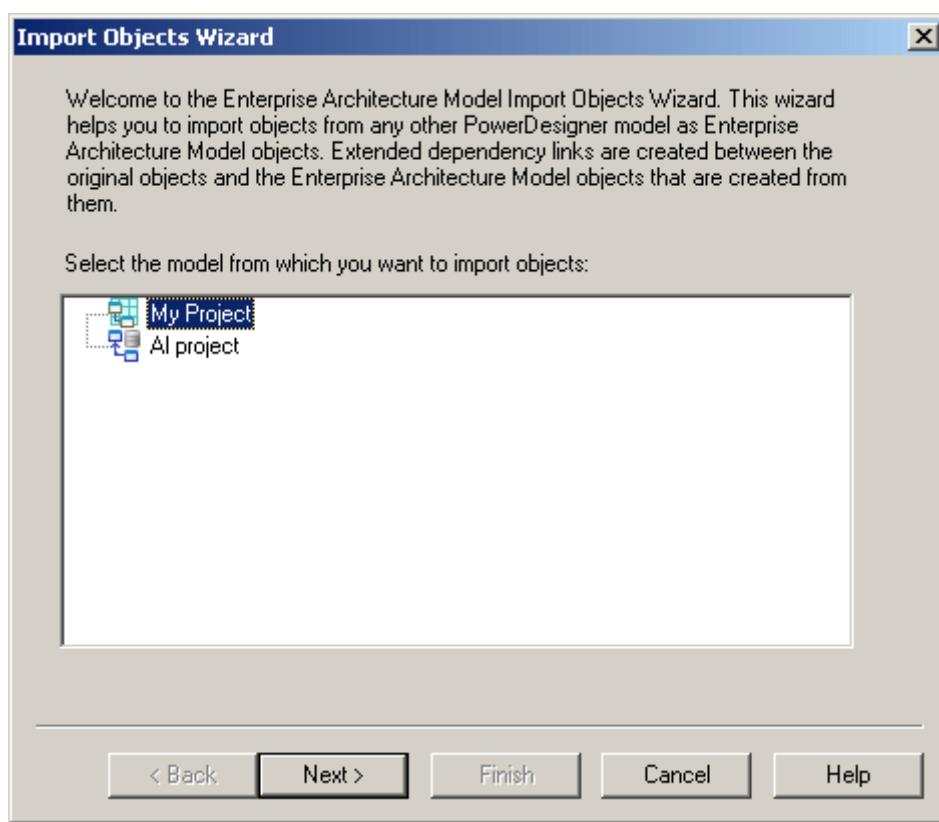
The EAM objects are exported to and created in the specified model.

10.3.2 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.

Procedure

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.

10.4 Traceability Links

Traceability links provide a flexible means for creating a connection between any object in any type of model and any other object in the same model or any other model of any type. Traceability links have no formal semantic meaning, but can be followed when performing an impact analysis or otherwise navigating through the model structure.

To create a traceability link between objects in the same diagram, select the *Link/Traceability Link* tool in the Toolbox. Click inside the symbol of the object that is dependent and, while continuing to hold down the mouse button, drag the cursor and release it on the symbol of the object on which it depends.

In the following example, the **Work** entity is shown as being dependent on **School** through a traceability link:



To create a traceability link to any object in any model that is open in the Workspace, open the property sheet of the dependent object, click its *Traceability Links* tab, and click the *Add Objects* tool. Use the *Model* list to select a different model, select the object to point to and click *OK* to create the link and return to the dependent object's *Traceability Links* tab.

You can optionally specify a type for any traceability link in the *Link Type* column.

Click the *Types and Grouping* tool to perform various actions on this tab:

- To make a link type available for selection in the *Link Type* column, click the *Types and Grouping* tool and select *New Link Type*. Enter a *Name* for the link type and, optionally, a *Comment* to explain its purpose, and then click *OK*.

i Note

Traceability link types created in this way are stored as stereotypes in an extension file embedded in the model. To work directly with this file click the *Types and Grouping* tool and select *Manage Extensions*. For detailed information about working with these files, see *Customizing and Extending PowerDesigner > Extension Files*.

- To control the display and grouping of links, click the *Types and Grouping* tool and select:
 - No Grouping* - to display all the links in a single list.
 - Group by Object Type* - to display links to different types of objects on separate sub-tabs. To add a link to a new object type, click the plus sign on the leftmost sub-tab.
 - Group by Link Type* - to display different link types on separate sub-tabs. To add a new link type, click the plus sign on the leftmost sub-tab.

i Note

To see all of the objects that point to an object via traceability links, open its property sheet, click its *Dependencies* tab, and click the *Incoming Traceability Links* sub-tab.

10.4.1 Importing Traceability Links from Excel

You can import traceability links, which can link any objects in your models with a semantic that you define, from lists in Excel files. An Excel file filled with sample data is provided in the PowerDesigner Examples folder, which you can adapt for your own imports.

Context

This file forms part of a sequence of files that can be used to create the example model EA Example. For information about the other files in the sequence, see [Importing EA Metadata from Excel \[page 126\]](#).

Procedure

1. To create a new EAM to import into, select **File** **Import** **Excel File**. Select *Enterprise Architecture Model* from the model type list, enter a **Model name**, and then click **OK** to open the Excel Import Wizard.

Note

You can import an Excel file into an existing model by right-clicking the model in the Browser and selecting *Import Excel File*. If this option is not available, the Excel Import extension is not attached to your model. To attach it, see [Extending your Modeling Environment \[page 25\]](#).

2. Click the **Select File** tool, browse to and select the Excel file to import, and click **Open** to return to the wizard.

The example file `09 Traceability Links Import.xlsx` contains three sheets, each of which links objects together via traceability links.

Note

Certain metadata in the file cannot be automatically evaluated by the wizard. This file cannot be imported using the *Auto-map columns to properties* option and you must specify the mappings in the wizard.

3. Click the **Options** button set the import options as follows, and click **OK** to return to the wizard:

- All options - deselected.
- Reference associated object by: Code**
- Qualified name separator: .** (dot)

For detailed information about these options, see *Core Features Guide > Modeling with PowerDesigner > Objects > Importing Objects from Excel Files*.

4. Click **Next** to begin the import.

Specify how your sheets and columns will be imported in the wizard as follows:

- Trace BF1 to Project** - Links first-level business functions to projects that they own via traceability links:
 1. Import the table as **Traceability Link**.

2. Import the column **DependentObject** as **Dependent Object** and set the *Object* type to **Business Function**.
 3. Import the column **LinkType** as **Link Type**.
 4. Import the column **InfluentObject** as **Influent Object** and set the *Object* type to **Project**.
- **Trace BF2 to System** - Links second-level business functions to systems on which they depend via traceability links:
 1. Import the table as **Traceability Link**.
 2. Import the column **DependentObject** as **Dependent Object** and set the *Object* type to **Business Function**.
 3. Import the column **LinkType** as **Link Type**.
 4. Import the column **InfluentObject** as **Influent Object** and set the *Object* type to **System**.
 - **Trace P4 to System** - Links fourth-level processes to systems on which they depend via traceability links:
 1. Import the table as **Traceability Link**.
 2. Import the column **DependentObject** as **Dependent Object** and set the *Object* type to **Process**.
 3. Import the column **LinkType** as **Link Type**.
 4. Import the column **InfluentObject** as **Influent Object** and set the *Object* type to **System**.

Progress is displayed in the *Output* window. When the import is complete, a dialog will appear showing how many objects have been created. Click **OK** to return to your model and review the imported objects.

10.5 Specifying Diagrams as Related Diagrams

You can use *related diagrams* to further define the behavior and implementation of objects and to view them from different angles and in terms of semantic relationships. You can associate any type of diagram open in the workspace with an object, including diagrams from other packages or models.

To specify that a diagram is a related diagram of an object, open the property sheet of the object and click the *Related Diagrams* tab. If this tab is not visible, click the **More** button. The following tools are available:

Table 62:

Tool	Description
	Properties - Opens the property sheet of the selected diagram.
	Open Diagram - Opens the selected diagram. <p>i Note</p> <p>If the object is not composite (and so does not have any child diagrams), you can navigate to the first related diagram in the list from an object symbol by pressing Ctrl and double-clicking the symbol. To navigate to any related diagram in the list, including for composite objects, right-click the symbol and select</p> <p>► Related Diagram ► <Diagram Name> ▶</p>

Tool	Description
	<i>Add Objects</i> - Opens an object selection dialog to select existing diagrams and add them to the list.
	<p><i>Create an Object</i> - Creates a new diagram, and adds to the list as a related diagram. You must specify the location (model) and type of diagram to create and give it an appropriate name.</p> <p>i Note To create a new diagram and associate it to the object from the object symbol, right-click the symbol and select ► <i>Related Diagram</i> ► <New> ▶.</p>

10.6 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.

Procedure

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.

Results

i 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*.

Important Disclaimers and Legal Information

Coding Samples

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. SAP does not warrant the correctness and completeness of the Code given herein, and SAP shall not be liable for errors or damages caused by the usage of the Code, unless damages were caused by SAP intentionally or by SAP's gross negligence.

Accessibility

The information contained in the SAP documentation represents SAP's current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP in particular disclaims any liability in relation to this document. This disclaimer, however, does not apply in cases of wilful misconduct or gross negligence of SAP. Furthermore, this document does not result in any direct or indirect contractual obligations of SAP.

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The SAP documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. SAP does not warrant the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP shall not be liable for any damages caused by the use of related information unless damages have been caused by SAP's gross negligence or willful misconduct. All links are categorized for transparency (see: <http://help.sap.com/disclaimer>).



www.sap.com/contactsap

© 2015 SAP SE or an SAP affiliate company. All rights reserved.
No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see <http://www.sap.com/corporate-en/legal/copyright/index.epx> for additional trademark information and notices.