

Môn học:

Công cụ Thiết Kế Hệ Thống Thông Tin

Mã MH: ITEC3407

Số TC: 3 (2LT, 1TH)

Khoa CNTT, ĐH Mở tp. HCM

CHƯƠNG 6: Mô hình XML và BÁO CÁO

- ◆ 6.1. MÔ HÌNH XML
 - ◆ 6.1.1. Cơ bản về mô hình XML
 - ◆ 6.1.2. Xây dựng mô hình XML
 - ◆ 6.1.3. Làm việc với mô hình XML
 - ◆ 6.1.4. Các mô hình được sinh ra từ MH HƯỚNG ĐT
- ◆ 6.2. TÀI LIỆU BÁO CÁO PT&TK với Power Designer
 - ◆ 6.2.1. Sử dụng trình soạn thảo báo cáo
 - ◆ 6.2.2. Quản lý các mô hình báo cáo (đơn và đa chiều)
 - ◆ 6.2.3. Xây dựng các báo cáo
- ◆ 6.3. Tổng kết chương & Bài tập

6.1. MÔ HÌNH XML

- ◆ 6.1.1. Cơ bản về mô hình XML
- ◆ 6.1.2. Xây dựng mô hình XML
- ◆ 6.1.3. Làm việc với mô hình XML
- ◆ 6.1.4. Các mô hình được sinh ra từ MH HƯỚNG ĐT

6.1.1. Cơ bản về mô hình XML

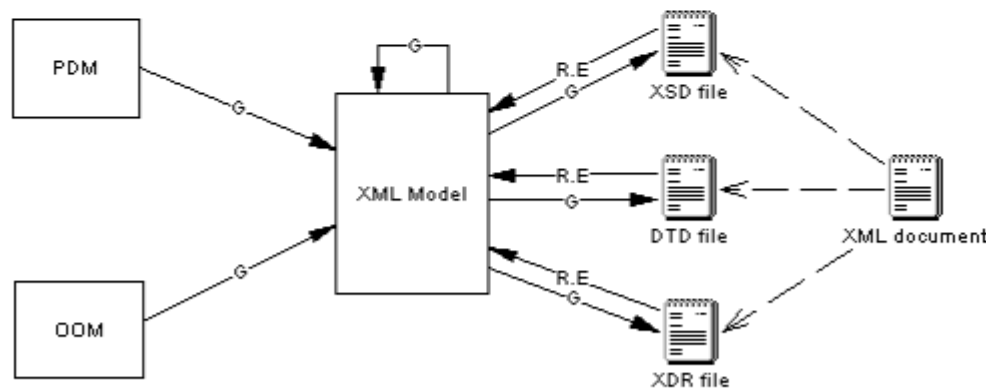
Functional overview

With the graphical interface and the Browser tree view of PowerDesigner XML Model, you can design an XML diagram which represents the content of an XML Schema Definition file (.XSD), a Document Type Definition file (.DTD) or an XML-Data Reduced file (.XDR). Since XML structures can be very complex, it is much easier to visualize them through comprehensive and explicit diagrams, than to read XML-coded pages.

6.1.1. Cơ bản về mô hình XML

The XML Model allows you to:

- ◆ Build an XML model
- ◆ Check an XML model
- ◆ Map objects in an XML model
- ◆ Edit a report of an XML model
- ◆ Generate and reverse engineer an XSD, a DTD or an XDR file
- ◆ Generate an XML model from a Physical Data Model (PDM)
- ◆ Generate an XML model from an Object Oriented Model (OOM)
- ◆ Generate an XML model from an XML model



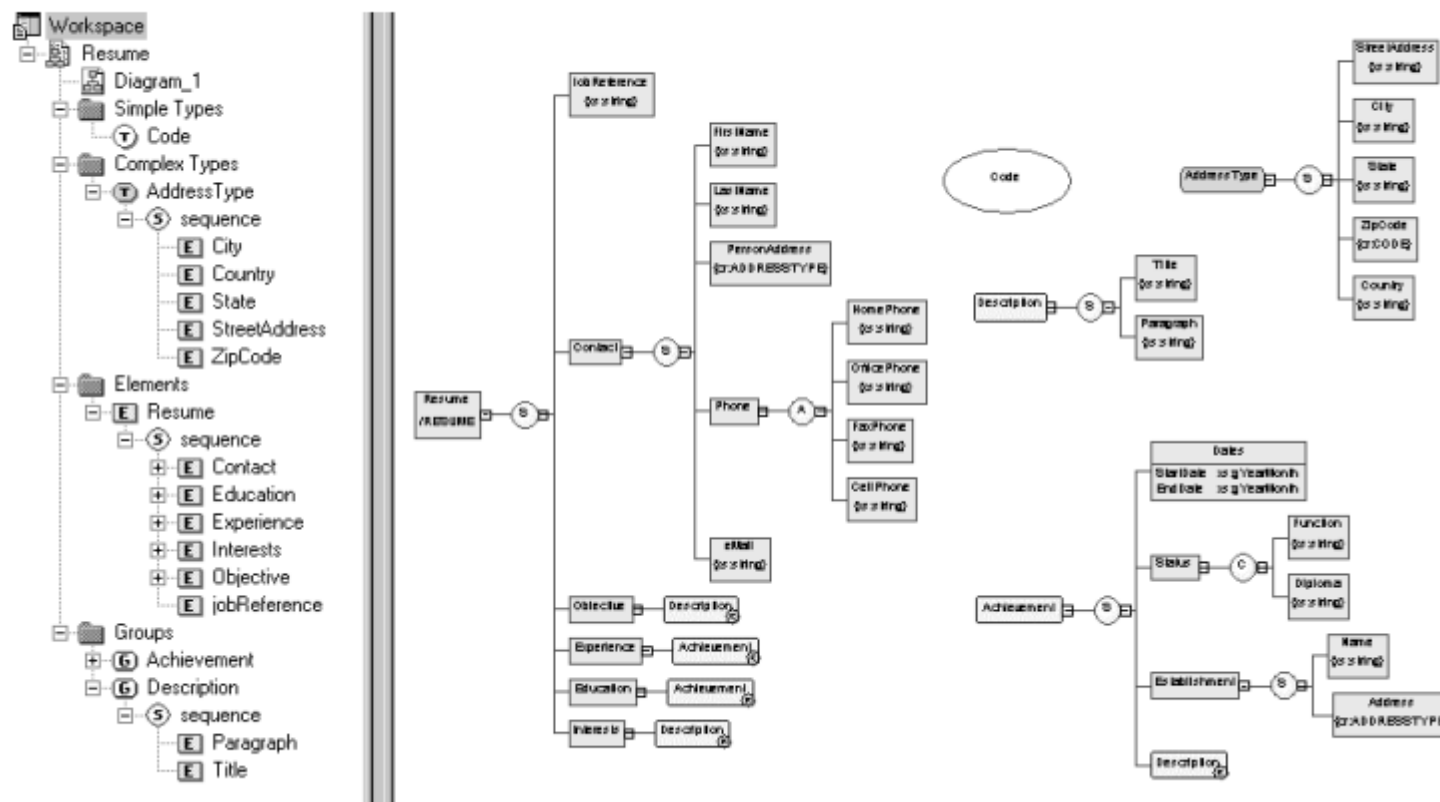
R.E: Reverse Engineering
G: Generation

6.1.1. Cơ bản về mô hình XML

What is an XML model?

An XML model is a graphical representation of an XML Schema Definition file (.XSD), a Document Type Definition file (.DTD) or an XML-Data Reduced file (.XDR).

Example of an XML model (Browser and diagram):

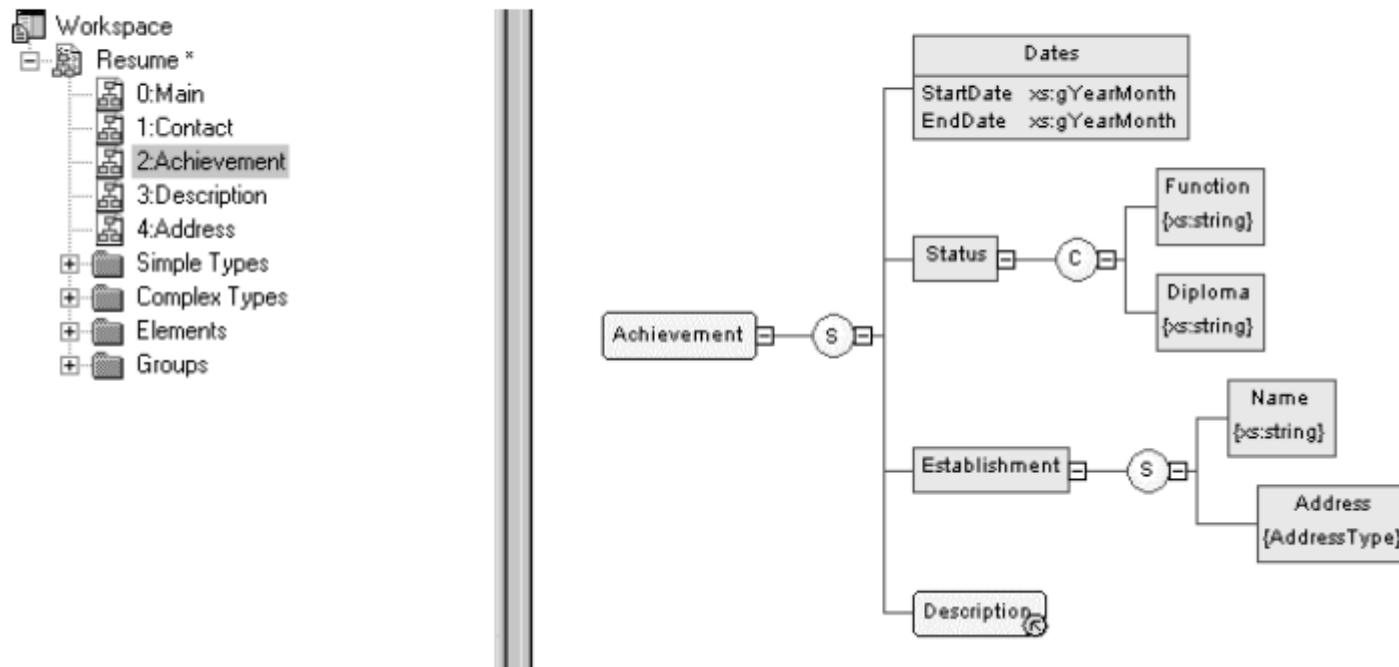


6.1.2. Xây dựng mô hình XML

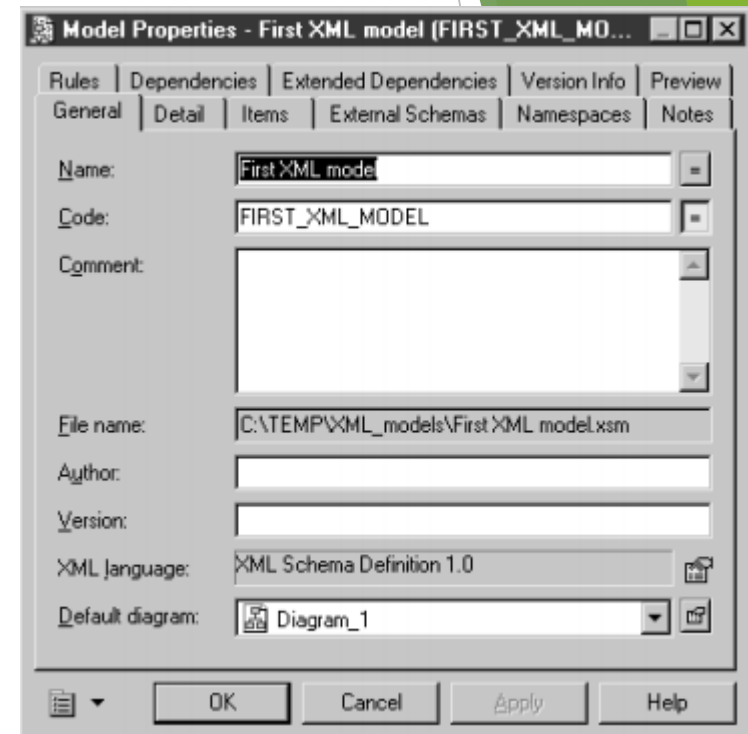
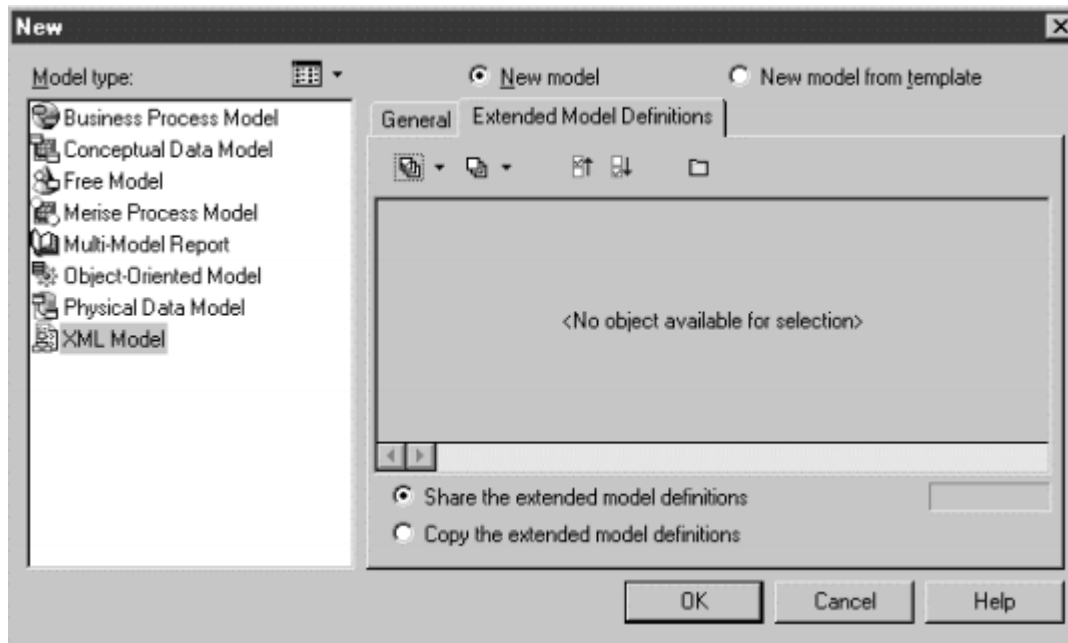
XML diagram basics

You can create XML diagrams in an XML model.

The following example displays the Achievement diagram of the Resume XML model:



6.1.2. Xây dựng mô hình XML



Creating an XML model

There are several ways to create an XML model:

- ◆ Create a new XML model
- ◆ Create a new XML model using a template
- ◆ Create an XML model using existing elements (reverse engineering an XSD, a DTD or an XDR file, generating from a PDM or an OOM)

6.1.2. Xây dựng mô hình XML

Why build an XML diagram?

An XML diagram is the easiest way to define the structure and content of an XML document if you are not familiar with the syntax of XML Schema Definition (XSD), Document Type Definition (DTD) or XML-Data Reduced (XDR).

With the user-friendly graphical interface of PowerDesigner XML Model, you can build an XML diagram and then generate automatically an XSD, a DTD or an XDR file.

6.1.2. X

Generated schema:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  <xs:element name="XML_DOCUMENT">
    <xs:complexType>
      <xs:choice>
        <xs:element name="XSD_FILE">
          <xs:complexType>
            <xs:sequence>
              <xs:element ref="XML_DECLARATION"/>
              <xs:element name="SCHEMA_TAG">
                <xs:complexType>
                  <xs:all>
                    <xs:element name="GLOBAL_ELEMENT_TAG"/>
                    <xs:element name="GLOBAL_ATTRIBUTE_TAG"/>
                    <xs:element name="GLOBAL_SIMPLETYPE_TAG"/>
                    <xs:element name="GLOBAL_COMPLEXTYPE_TAG"/>
                  </xs:all>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="DTD_FILE">
          <xs:complexType>
            <xs:sequence>
              <xs:element ref="XML_DECLARATION"/>
              <xs:element name="DOCTYPE_TAG">
                <xs:complexType>
                  <xs:all>
                    <xs:element name="ELEMENT_TAG"/>
                    <xs:element name="ATTLIST_TAG"/>
                    <xs:element name="ENTITY_TAG"/>
                  </xs:all>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
  </xs:element>
  <xs:element name="XML_DECLARATION">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="VERSION_ATTRIBUTE"/>
        <xs:element name="ENCODING_ATTRIBUTE"/>
        <xs:element name="STANDALONE_ATTRIBUTE"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

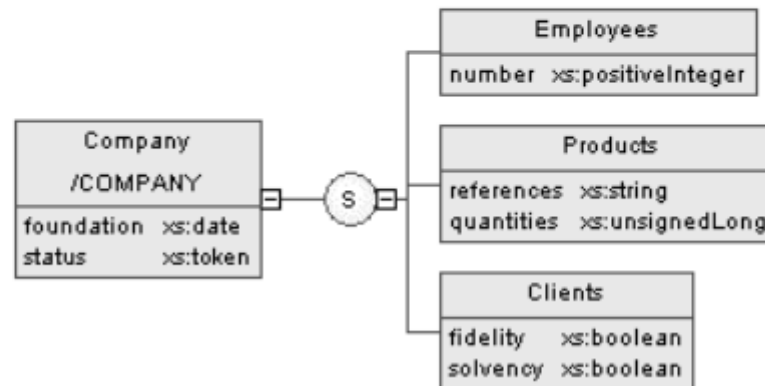
In a schema, elements are declared with `<element>` tags.

6.1.2. Xây dựng mô hình XML

Defining the attributes of an element

Attributes are used to give additional information about elements.

For example:



6.1.2. Xây dựng mô hình XML

Generated schema:

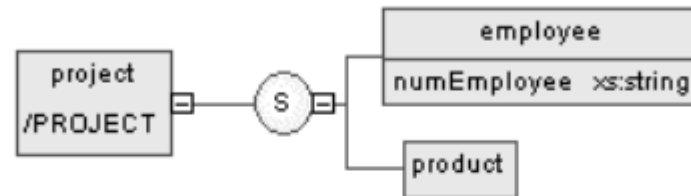
```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="COMPANY">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="EMPLOYEES">
          <xs:complexType>
            <xs:attribute ref="NUMBER">
            </xs:attribute>
          </xs:complexType>
        </xs:element>
        <xs:element name="PRODUCTS">
          <xs:complexType>
            <xs:attribute name="REFERENCES" type="xs:string">
            </xs:attribute>
            <xs:attribute name="QUANTITIES" type="xs:unsignedLong">
            </xs:attribute>
          </xs:complexType>
        </xs:element>
        <xs:element name="CLIENTS">
          <xs:complexType>
            <xs:attribute name="FIDELITY" type="xs:string">
            </xs:attribute>
            <xs:attribute name="SOLVENCY" type="xs:string">
            </xs:attribute>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
      <xs:attribute name="FOUNDATION" type="xs:date">
      </xs:attribute>
      <xs:attribute name="STATUS" type="xs:token">
      </xs:attribute>
    </xs:complexType>
  </xs:element>
  <xs:attribute name="NUMBER" type="xs:positiveInteger">
  </xs:attribute>
</xs:schema>
```

6.1.2. Xây dựng mô hình XML

Defining a unique constraint

A unique constraint specifies that an element or an attribute value (or set of values) must be unique or null within a specified scope.

For example:



Generated schema:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="PROJECT">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="EMPLOYEE">
          <xs:complexType>
            <xs:attribute name="NUMEMPLOYEE" type="xs:string">
            </xs:attribute>
          </xs:complexType>
        </xs:element>
        <xs:element name="PRODUCT"/>
      </xs:sequence>
    </xs:complexType>
    <xs:unique name="UNIQUENUM">
      <xs:selector xpath="employee"/>
      <xs:field xpath="@numEmployee"/>
    </xs:unique>
  </xs:element>
</xs:schema>
```

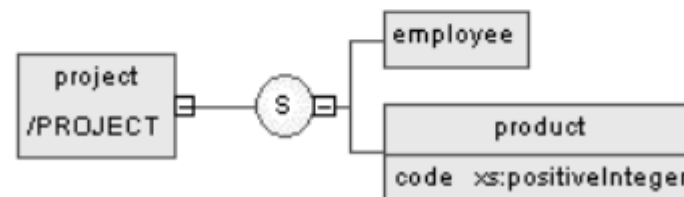
6.1.2. Xây dựng mô hình XML

Defining a key constraint

A key constraint specifies that an element or an attribute value (or set of values) must be a key within a specified scope.

A key means that data should be unique, not null and always present within a specified scope.

For example:



Generated schema:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="PROJECT">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="EMPLOYEE"/>
        <xs:element name="PRODUCT">
          <xs:complexType>
            <xs:attribute name="CODE" type="xs:positiveInteger"/>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
    <xs:key name="KEYCODE">
      <xs:selector xpath="product"/>
      <xs:field xpath="@code"/>
    </xs:key>
  </xs:element>
</xs:schema>
```


6.1.2. Xây dựng mô hình XML

XPath abbreviated syntax

You can use the following abbreviated syntax to define an XPath expression:

Syntax	Description
/	Root node of the XML document. It is the root element with its ramifications
.	Selects the context node. It is the current element (on which an identity constraint is defined) with its ramifications
..	Selects the context node parent
*	Selects all the child elements of the context node
employee	Selects all the employee child elements of the context node
s:employee	Selects all the employee child elements of the context node, defined in the namespace with the “s” prefix
@numEmployee	Selects the numEmployee attribute of the context node

6.1.2. Xây dựng mô hình XML

Defining notations

Notations allow you to define and process non-XML objects within an XML model.

For example: picture files with a .GIF extension.

Generated schema:

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <!--
    Integrating GIF files in your XML model
  -->
  <xs:notation name="PICTURES" public="pictures/gif"
    system="user/local/pictureViewer"/>
</xs:schema>
```

Notations are not available on models targeted with XDR.

6.1.3. Làm việc với mô hình XML

Checking an XML model

The XML Model is a very flexible tool. It should allow you to build your model without any controls or constraints on data exchange and consistency of the system.

However, you can use the Check Model feature at any time to control the consistency and correctness of the model you are building.

You can:

- ◆ Define check options, such as level of problem severity and automatic correction
- ◆ Select objects to be verified
- ◆ Check an XML model
- ◆ Reuse check options previously defined
- ◆ Make corrections based on XML model check results

6.1.3. Làm việc với mô hình XML

Mapping objects in an XML model

Object mapping is the ability to establish a correspondence between objects belonging to heterogeneous models and diagrams.

The following table shows all the allowed mappings between XML Model objects and PDM or OOM objects:

XML Model object	PDM object	OOM object
Element	Table, view, column, view column	Class, class attribute
Element attribute	Column, view column	Class attribute
Complex type	Abstract data type	Class
Complex type attribute	Abstract data type attribute	Class attribute

6.1.3. Làm việc với mô hình XML

Understanding object mapping

You create a mapping between XML Model objects and PDM or OOM objects to setup a structure for data movement and transformation. Data comes from a data source and is loaded in an XML model.

In an XML model, the data source can be a PDM representing a database or an OOM representing classes.

Object mapping consists in linking objects in a PDM or an OOM data source to objects in an XML model.

6.1.3. Làm việc với mô hình XML

Manipulating XML objects graphically

The graphical interface of PowerDesigner allows you to manipulate XML objects within or between the Browser tree view and the diagram window.

A global object is right under the model item in the Browser tree view. It has no parent symbol in the diagram.

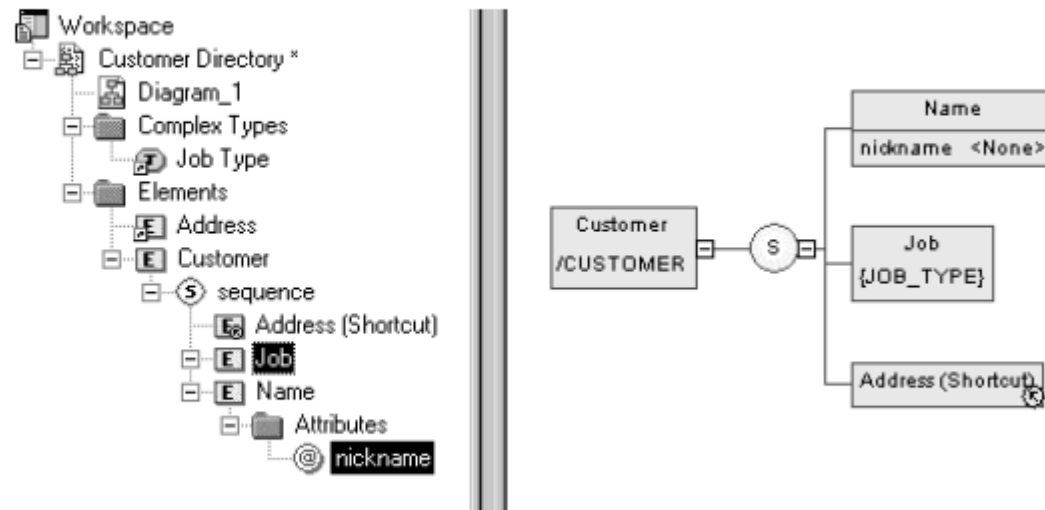
A local object is under a group particle item in the Browser tree view. It has a parent symbol in the diagram.

6.1.3. Làm việc với mô hình XML

Example: converting a local object into a global object

If the new global object does not appear in the diagram, select Symbol→Show Symbols and click the corresponding tab to select the object symbol.

Example before
local to global
conversions within
the Browser



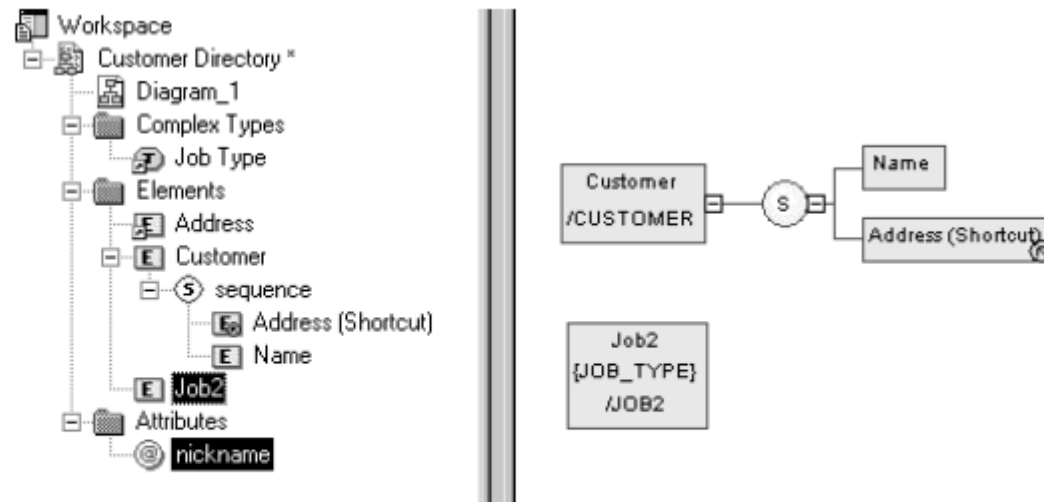
Job is a child element of the Customer element.

Nickname is the attribute of the Name element.

6.1.3. Làm việc với mô hình XML

Example: converting a local object into a global object

After conversions



Job became Job2, a global element.

Nickname became a global attribute.

6.1.3. Làm việc với mô hình XML

Comparing and merging XML models

You can compare and merge two XML models with the same XML language.

The comparison process allows you to highlight the differences between two XML models.

The merge process allows you to form a single model that combines design efforts performed independently by several team members.

6.1.3. Làm việc với mô hình XML

Generating an XML model from a Physical Data Model

Generating XML Model objects

When you generate an XML model from a PDM, PowerDesigner translates PDM objects into specified XML Model objects as follows:

PDM object	Generated object in an XML model
Domain	Simple type
Table	Element. A table is generated as a child element when it has one outgoing reference link with the Mandatory parent property selected
Column	Element or element attribute (See generation options)
View	Element
View column	Element attribute
Key	Key
Index	Unique
Abstract data type	Complex type
Abstract data type attribute	Complex type attribute

6.1.3. Làm việc với mô hình XML

Generating an XML model from an Object-Oriented Model

Generating XML Model objects

When you generate an XML model from an OOM, PowerDesigner translates OOM objects into specified XML Model objects as follows:

OOM Object	Generated object in an XML model
Class	Element. A class is generated as a child element when it has one composition link with another class
Class attribute	Element or element attribute (See generation options)
Identifier	Key
Domain	Simple type
Abstract class	Complex type
Abstract class attribute	Complex type attribute

6.1.3. Làm việc với mô hình XML

Editing an XML model report

You can edit a report of your XML model.

The Report Editor allows you to use a predefined report template or to build your own report.

What is a report?

A report is an easy to consult document that shows parts or the global content of a model. You can print a report in order to have it on a paper support, or you can generate it in RTF or HTML format for a better reusability.

Creating an XML model report

You create an XML model report using the Report Editor. You cannot open the Report Editor without at least one model opened in the workspace.

When you create a model report, you need the following information:

Option	Description
Report name	Name of the report provided by default
Language	Language in which the report can be printed. English is the default language
Report template	List of available templates related to the opened model

6.1.4. Các mô hình được sinh ra từ MH XSM

Generation basics

You can generate an XML model from an XML model.

There are two options to generate an XML model from an XML model:

Generation option	Description
Generate new XML model	It creates a copy of the source XML model, converting the source language (DTD, XSD or XDR) into the target language
Update existing XML model	It creates a default model with the objects translated from the XML model that is merged with an existing XML model. You can update, delete or add objects in the existing XML model (right pane) based on modifications made in the default model (left pane)

6.1.4. Các mô hình được sinh ra từ MH XSM

Generating an XML model from an XML model

Why generate an XML model from an XML model?

You can generate an XML model from an XML model when you need to keep two models synchronized during the design process.

This kind of generation allows you to create a copy of a given model and define generation links between objects in the source XML model and their equivalent in the generated XML model. When changes are made to the source model, they can then be easily propagated to the generated models using the Update Existing Model generation mode.

The generated model is the one that usually contains more information.

Generating and updating an XML model

The General page of the XML Model Generation Options dialog box displays the following options:

- ◆ Generate new XML model
- ◆ Update existing XML model

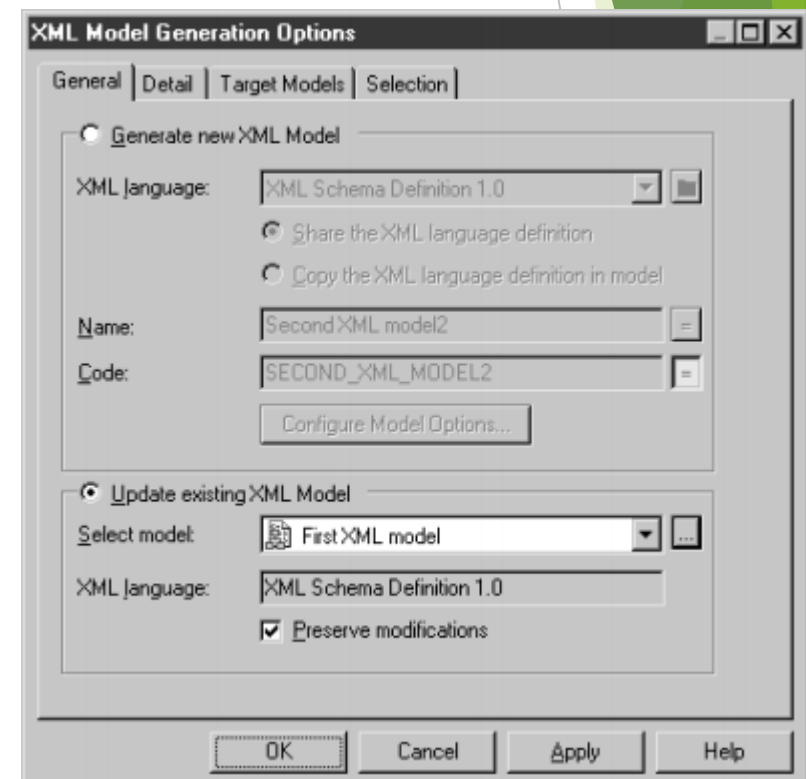
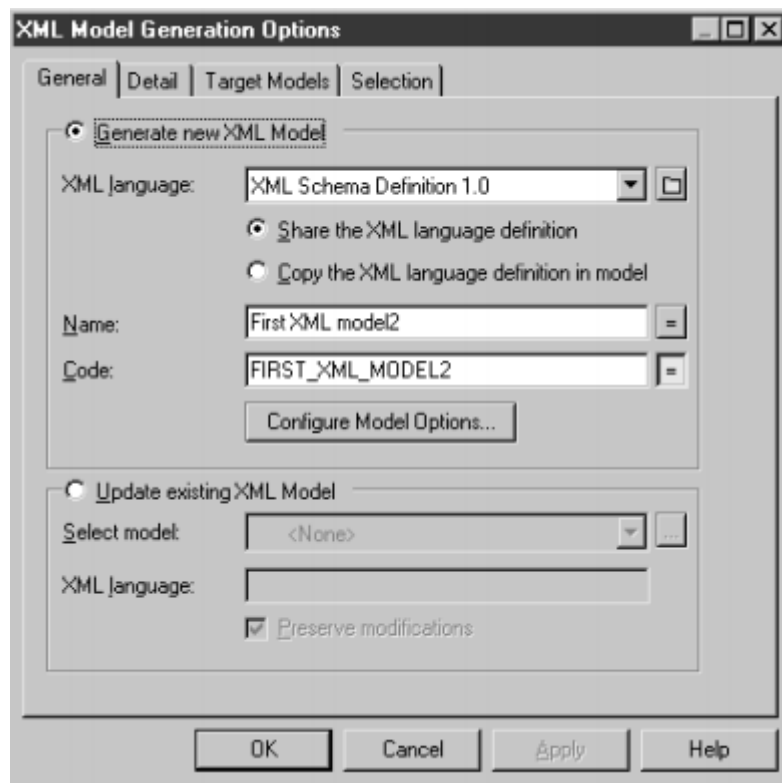
6.1.4. Các mô hình được sinh ra từ MH XSM

Generating an XML model from an XML model

Generating and updating an XML model

The General page of the XML Model Generation Options dialog box displays the following options:

- ◆ Generate new XML model
- ◆ Update existing XML model



6.2. TÀI LIỆU BÁO CÁO PT&TK với Power Designer

- ◆ 6.2.1. Sử dụng trình soạn thảo báo cáo
- ◆ 6.2.2. Quản lý các mô hình báo cáo (đơn và đa chiều)
- ◆ 6.2.3. Xây dựng các báo cáo

6.2.1. Sử dụng trình soạn thảo báo cáo

- ◆ Báo cáo là gì?
- ◆ Giới thiệu về Report Editor
- ◆ Sử dụng về template trong Report Editor
- ◆ Sử dụng Report Editor
- ◆ Organize với Report Editor
- ◆ Sử dụng công cụ về ngôn ngữ trong Report Editor

6.2.1. Sử dụng trình soạn thảo báo cáo: Báo cáo là gì?

What is a report?

A **report** allows you to publish valuable information about one model (model report) or several models (multi-model report) using different outputs. You can print a report in order to have it on a paper support, or you can generate it in RTF or HTML format for better reusability.

6.2.1. Sử dụng trình soạn thảo báo cáo: Giới thiệu về Report Editor

Introducing the Report Editors

Report Template Editor

The **Report Template Editor** is an independent module. It allows template creation and modification. You can open the Report Template Editor with no model attached to it.

A **report template** is an independent and reusable file saved on your hard disk, which indicates what information to include in your report.

Report Editor

You use the **Report Editor** to define a report structure in order to generate a report for a **single model**.

A model report is the association of a model, a report structure and a selection of objects. It graphically reflects the structure of a model.

Multi-Model Report Editor

You use the **Multi-Model Report Editor** to define a report structure in order to generate a **multi-model report**.

6.2.1. Sử dụng trình soạn thảo báo cáo: Sử dụng về template trong Report Editor

Using the Report Template Editor

You use the Report Template Editor to build report templates.

With the Report Template Editor, you can:

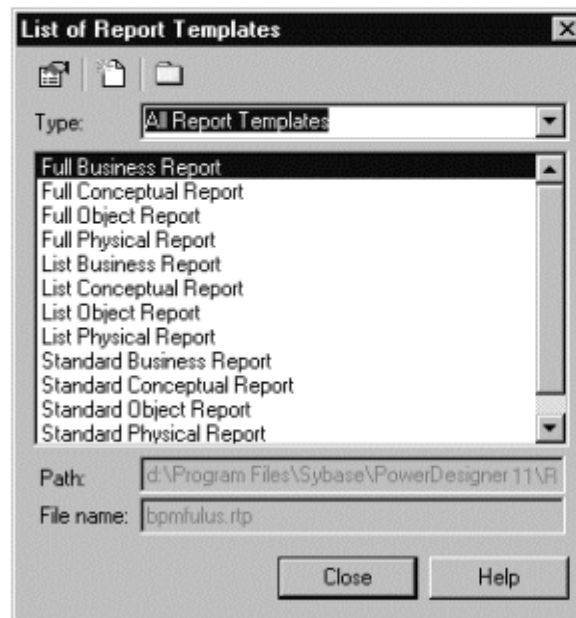
- ◆ Create a template
- ◆ Modify a template
- ◆ Save a template in a .RTP file

6.2.1. Sử dụng trình soạn thảo báo cáo: Sử dụng về template trong Report Editor

❖ To open the Report Template Editor:

- 1 Select Tools→Resources→Report Templates.

The List of Report Templates appears. It displays the alphabetical list of all the templates saved on your hard disk.



- 2 Select a template type in the Type dropdown listbox.

6.2.1. Sử dụng trình soạn thảo báo cáo: **Sử dụng Report Editor**

Using the Report Editor

You use the Report Editor to build report structures. A report structure indicates what will be printed in a report.

With the Report Editor, you can create and modify the structure of a report. You can include objects from the model diagram or listed objects in a report structure.

The Report Editor allows you to preview and print a report and also generate a report as an RTF or HTML file.

Opening the Report Editor from a model

You open the Report Editor from a model using the List of Reports.

From this list, you can also create, modify or delete a report. You can preview a report, print it or generate it as RTF or HTML file.

❖ To open the Report Editor from a model:

- 1 Select Model→Reports.

6.2.1. Sử dụng trình soạn thảo báo cáo: **Organize** với **Report Editor**

Organizing the Report Editor

A report must always contain at least one **section**, which appears as a tab at the bottom of the Report Items pane of the Report Editor.

A report section allows you to segment your model in order to facilitate your analysis. Each section has its own selection of objects and can only be of one type (PDM, CDM, OOM, BPM, XSM or FEM).

6.2.1. Sử dụng trình soạn thảo báo cáo: **Language với Report Editor**

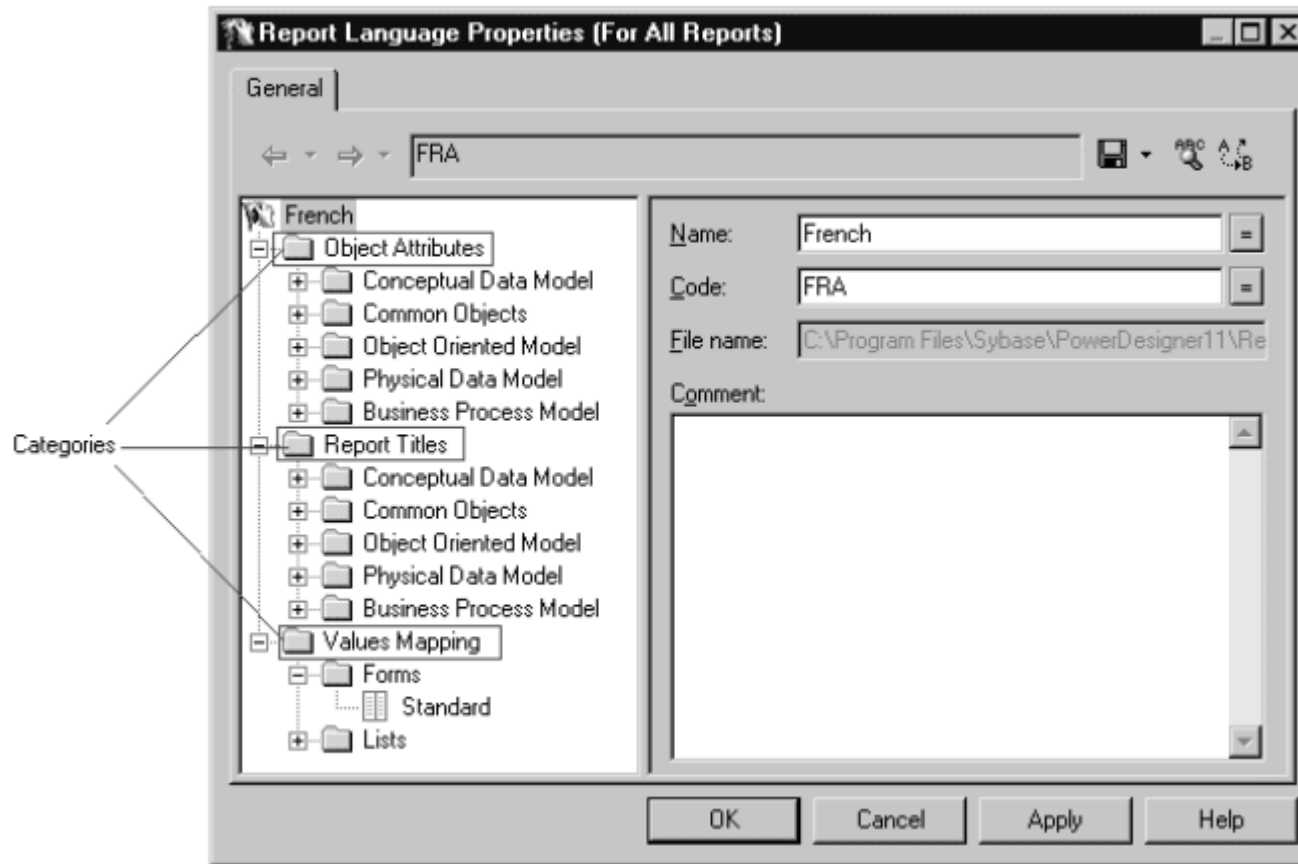
Using the Report Language Editor

You use the Report Language Editor to create or modify report language resource files.

Report language resource file

A report language resource file is a file in XML format saved with the .XRL extension. It contains all the printable texts of a report and their default values. You use it when you create your report and later generate it whatever the output. A report language resource file is stored in a central area and can be shared by any report to guarantee data consistency and save time to the user.

6.2.1. Sử dụng trình soạn thảo báo cáo: **Language với Report Editor**



6.2.2. Quản lý các mô hình báo cáo

- ◆ Báo cáo đơn chiều
- ◆ Báo cáo đa chiều

6.2.2. Quản lý các mô hình báo cáo: Báo cáo đơn chiều

Using a report template for a model report

You can use a template to generate a report for a selected model. Report templates are reusable, they list information to include in reports.

You can create your own template in a given language, modify it and save it in an RTP file. However, when you use a template created in a language different from the one you select to create your report, only user-defined items such as Title or Text paragraph will keep the language of the template. Other items will be displayed in the report language.

6.2.2. Quản lý các mô hình báo cáo: Báo cáo đơn chiều

Creating a model report

You create a model report using the Report Editor. You cannot open the Report Editor without at least one model opened in the workspace.

When you create a model report, you need the following information:

Option	Description
Report Name	Name of the report provided by default
Language	Language in which the report can be printed. English is the default language
Report Template	List of available templates related to the opened model: <ul style="list-style-type: none">◆ Full: contains the table of contents and all main model items◆ Standard: contains the table of contents, model and package graphs and most list items◆ List: contains Title item and all list items

6.2.2. Quản lý các mô hình báo cáo: Báo cáo đơn chiều

Generating a model report

When you generate a model report, you must select a destination file (RTF, HTML, or Print)

Depending on the generation option you select, you generate your report on a printed paper or in a file:

Generation option	Output
RTF	RTF file
HTML	HTML file
Print	Printed paper

Modifying a model report

You can use an existing model report to generate a modified model report.

You can select an existing report from the List of Reports if you need to:

- ◆ Generate for a different output
- ◆ Change selection of objects
- ◆ Add sections

6.2.2. Quản lý các mô hình báo cáo: Báo cáo đơn chiều

Saving a model report

When you want to save a model report, you have to save the model.
You can then keep the report for a later use.

6.2.2. Quản lý các mô hình báo cáo: Báo cáo đa chiều

Using a report template for a multi-model report

You can use templates to generate a multi-model report. Report templates are reusable, they list information to include in reports.

A multi-model report can contain items of all the different types of models of your session. It allows you to have a global view across different models.

Creating a multi-model report

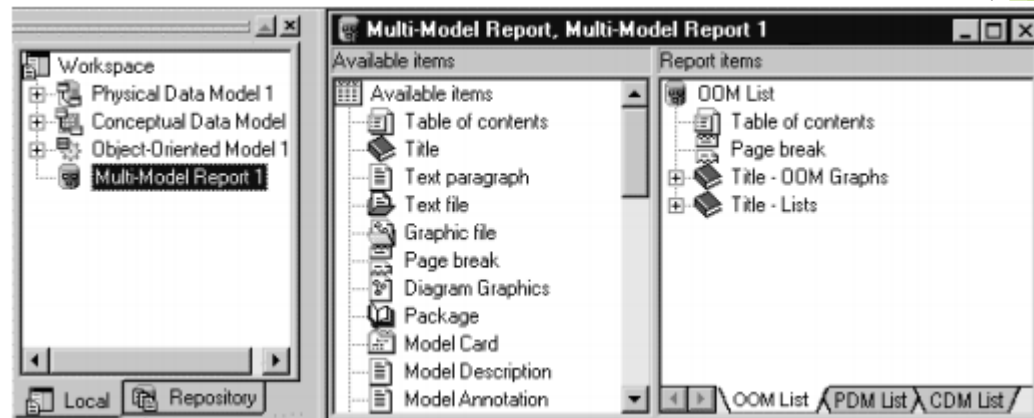
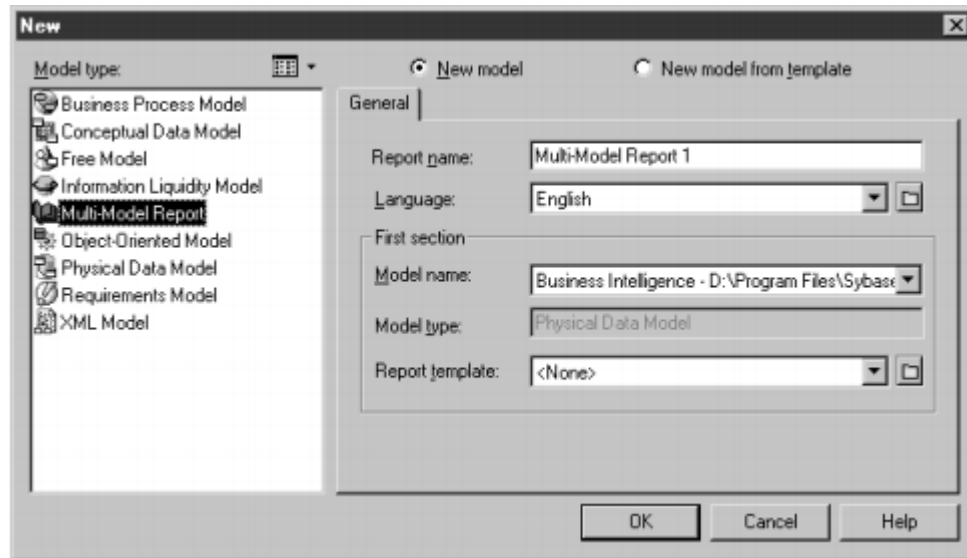
You create a multi-model report using the Multi-Model Report Editor.

You cannot create a multi-model report without at least one model opened in the workspace.

At creation, a multi-model report must always contain a section that you define by selecting a model.

For example, if you want the first section of your multi-model report to be an OOM type section, you have to select an OOM model in the New Multi-Model Report dialog box.

6.2.2. Quản lý các mô hình báo cáo: Báo cáo đa chiều



6.2.2. Quản lý các mô hình báo cáo: Báo cáo đa chiều

Generating a multi-model report

When you generate a multi-model report, you must select a destination file (RTF, HTML, or Print)

Depending on the generation option you select, you generate your report on a printed paper or in a file.

Generation option	Output
RTF	RTF file
HTML	HTML file
Print	Printed paper

You can customize your HTML report before generate it.

6.2.3. Xây dựng các báo cáo

- ◆ Các thành phần báo cáo (report items)
- ◆ Các node (nodes)
- ◆ Các thành phần (items)
- ◆ Các thành phần định dạng (formatting items)
- ◆ Thiết lập trang báo cáo
- ◆ Print preview

6.2.3. Xây dựng các báo cáo: report items

Managing report items

You can use the following types of items in your report:

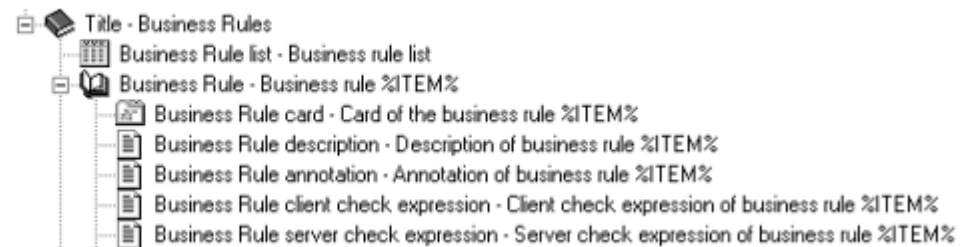
Item	Description
Node	Contains dependent items. Nodes appear as books in the Report Editor
Model-dependent item	Contains information about a model or a package
Object list	Contains tables listing objects from a model or a package
Object-dependent item	Contains information about a specific object type and can only appear under an object node
Independent item	Can appear anywhere in a report (for example, a table of contents or a graph)

6.2.3. Xây dựng các báo cáo: **nodes**

Using nodes

A node is an item that contains dependent items. Nodes appear as books in the Report Editor.

For example the Business rule node contains items that describe a business rule.



6.2.3. Xây dựng các báo cáo: **items**

Using items

In a report, items can be of the following types:

- ◆ Model-dependent items contain information about a model or a package
- ◆ Object-dependent items contain information about a specific type of object
- ◆ Independent items do not contain information about specific models or objects

6.2.3. Xây dựng các báo cáo: **items**

Model-dependent items

Model-dependent items provide information about a model or a package. If you insert a model-dependent item under a package node, the item contains information related to the package.

You cannot insert model-dependent items into object nodes.

The following are model-dependent items:

Dependent item	Print format	Content
Card	List of properties	Properties of the model or package
Description	Text paragraph	Description of the model or package
Annotation	Text paragraph	Annotation of the model or package
Begin script (PDM only)	Text paragraph	Text to add at the beginning of a script generated from the model
End script (PDM only)	Text paragraph	Text to add at the end of a script generated from the model
Object list	Table	Table that lists all objects of a given type in the model

6.2.3. Xây dựng các báo cáo: **formatting items**

Formatting items

In a report, the available format options depend on the print format:

- ◆ Text format
- ◆ Graph format
- ◆ Table format

6.2.3. Xây dựng các báo cáo: report page

Setting up report pages

Before you print a report, you can set up its pages as follows:

- ◆ Include a header and footer in each report section
- ◆ Include a title page for the entire report

These options do not correspond to items in the Report Items pane.

6.3. Tổng kết chương & Bài tập

Hãy vẽ mô hình XSM cho mô tả hệ thống. Và tạo sinh báo cáo.

(sẽ được cung cấp tại lớp học)

Môn học:

Công cụ Thiết Kế Hệ Thống Thông Tin

Giảng viên: ThS. VÕ THỊ KIM-ANH

(2021)