



ARIS Education Package

ARIS Architect Beginner Modeling Exercises-
University Edition

Version 2.2 | April, 2016



Software AG
University Relations

TABLE OF CONTENTS

Introduction.....	3
United Motor Group (UMG) scenario	7
Exercise 1 - Process Modeling Overview	8
Exercise 2 - Group Structure	9
Exercise 3 - Organizational Modeling	10
Exercise 4 - Organizational Modeling /Attribute Editing	12
Exercise 5 - Nested Objects in an IT Infrastructure Application System Type	13
Exercise 6 - Occurrence and Definition Level.....	15
Exercise 7 - Create Occurrence Copy /Organizational Responsibility in VACD	16
Exercise 8 - Incorrectly Event-driven Process Chain (EPC)	17
Exercise 9 - Process Overview with Even-Driven Process Chain (EPC)	19
Exercise 10 - Assignments (existing model)	20
Exercise 11 - Occurrence Copy / Define Organizational Responsibility in EPC	21
Exercise 12 - Detailed EPC /Assignments /Process Interfaces /Attribute Editing...	22
Exercise 13 - Detailed EPC /Assignments /Occurrence Copy	24
Exercise 14 - Modeling an Additional Detail EPC	27
About	

Introduction

Target Group

ARIS Education Package University Edition tutorial is aimed at students who do not have any specific knowledge of Process Model and Design, and who would like to teach themselves how to design, implement and simulate a business process.

Objectives

- Introduction to Business Process Analysis and Management
- Business Process Lifecycle
- Modules of ARIS Architect
- Advanced Modeling functionality
- Modeling with ARIS Architect: Creating library models and process models
- Introduction to ARIS Methodology
- Testing and evaluating models

Procedure

The tutorial comprises successive exercises that build upon each other. As you work through them, the configuration is extended step by step.

Database and other related material

All the documents and database required for the tutorial can be downloaded from the tutorials page of [ARIS community](#).

Go to the BPM Exercises section and download the “BPM Exercises with ARIS: Database & other related material”.

ARIS Community

The [ARIS community](#) contains all manuals and technical references available for AEP - PD&A project. It also provides the ARIS online help topics. These are usually only accessible via the ARIS user interface.

Sequence of reading

To make the best use of the **AEP - PD&A University Edition tutorial**, we recommend reading the documents supplied and performing the exercises in the following order:

- AEP - PD&A Installation Guide
- AEP - PD&A: ARIS Architect Beginner Modeling Exercises- University Edition
- AEP - PD&A: ARIS Architect Advanced Modeling Exercises - University Edition
- AEP - PD&A: ARIS Architect Administration Exercises - University Edition

Sample solution

The AEP - PD&A Architect Beginner Modeling Exercises - University Edition comes with a sample solution which has possible solution to each of the exercises in this tutorial. This solution document is available only to faculty members.

The solution document should not be referred to as the only correct solution to the AEP - PD&A: ARIS Architect Beginner Modeling Exercise. Please talk to your professor to get access to the sample solution.

Important Note

All the Exercises in AEP - PD&A: ARIS Architect Beginner Modeling Exercises- University Edition has to be done using the **United Motor Group** database.

- Start **ARIS Architect & Designer**
- Switch to the Explorer tab. Right-click on **United Motor Group** database
- Use the "Log in with options" and use Filter: "Entire method", Version context: "Workspace"

If the UMG database doesn't already exist in ARIS Architect, you can find it in the **BPM Exercises with ARIS: Database & other related material** you downloaded previously (see introduction section)

- Start **ARIS Architect & Designer**
- Switch to the Explorer tab. Right-click on **Local-> Restore Database**
- Chose the location of the United Motor Group.adb file (AEP_PDA_Materials\United Motor Group\United motor Group.adb)
- From the drop down select Restore as "Non Versionable database" and click on "Finish"

Information on how to use this guide

This chapter provides information on how to use this manual. In this manual, menu items, file names, etc. are identified by the following notational and formatting conventions.

Notation/Formatting	Example
Menu items, key combinations, dialog boxes, file names, model types, model names etc. are shown in bold .	Enter system as the user name and manager as the password. Click on Help to access the Help menu items and open further information sources.
Entries via the keyboard, menu item selections are shown within double quotes " ".	Enter "system" as the user name and "manager" as the password. Select Model Types: "EPC" and Object Types: "Event", "Function".
Connection types are shown in <i>italics</i> and are <u>underlined</u> .	This symbol uses the connection type <u><i>is predecessor of</i></u> .
Sequences of menus and menu items are in bold and separated by forward slashes.	Right-click on the chart background and select Display options/Show absolute indicator values .
Sequence of actions and selections are separated by an arrow→ and are in bold.	Go to Configuration→ Method → Model types.
Entries with user-defined content are highlighted in bold type and enclosed in angle brackets.	Enter the path <Directory>\PPM.
One-line example texts, e.g. a long directory path, that need to be shown on several lines due to a lack of space are separated by the ↵ (arrow) symbol at the end of the line.	D:\Program Files\ppm\↵ xml\Adapter

Particular text sections are specially identified:

Warning



Warnings indicate important instructions, where loss of data, semantical error or a safety risk could arise if they are not observed carefully.

Note

Notes will provide you with additional information.

Tip

Tips indicate references to explanatory notes, e.g. in manuals or in the online help.

See also

This refers to further information on the same topic or related topics within the document.

Hint



Hints provide additional information to optimize your results.

United Motor Group (UMG) scenario

United Motor Group, a global automaker, has teamed up with Software AG consultants to launch an ARIS Value Engineering project, using the business process transformation method. After assessing UMG's strategic positioning and defining objectives, the team identified critical success factors, which need to be improved to achieve the defined strategy. Streamlining the business processes is the key to fulfilling this objective.

UNITED MOTOR GROUP FACT & FIGURES

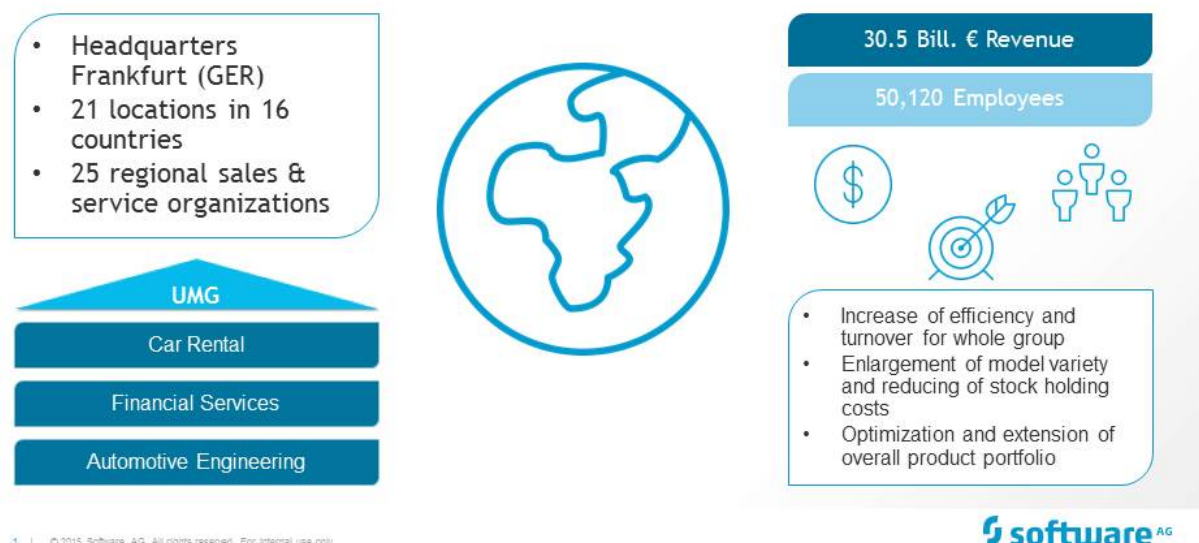


Figure 1 - Profile of United Motor Group

To create the necessary transparency with regard to business workflows, organizational structures and system landscapes, **ARIS Architect** has been selected as the modeling platform.

Note

Before getting started with the exercises, please make sure that you have read all the previous sections thoroughly.

Exercise 1 – Process Modeling Overview

The major automobile manufacturer United Motor Group has decided on a pilot project to optimize its processes with ARIS. The goal of this project is to first document all the processes in Direct Car Sales, forming the basis for further process documentation and optimization activities for later use.

In this exercise, you are to create a model, utilize the Mini Toolbar, familiarize yourself with the **Value-Added Chain Diagram (VACD)** model type and learn to maintain text descriptions in model attributes.

Procedure

- 1) Start **ARIS Architect & Designer**.
- 2) Switch to the Explorer tab and log in to the **United Motor Group** database. Use “Log in with options” and use the Filter: “Entire Method”, Version context: “Workspace”. Right-click on the **UMG** folder of your database and select **New→Model**. Create a VACD and name it “**Core Business Processes**”.



- 3) Model the **Core Business Processes VACD** as described below:
 - a) **Research and Development** begins the process which is then followed by **Production Development** which uses the connection type is predecessor of.
 - b) **Marketing** comes next which in-turn triggers **Sales**. The **Sales Function** is then followed by **Production Planning** and **Customer Care** running simultaneously.
- 4) Save and close the model.

Exercise 2 – Group Structure

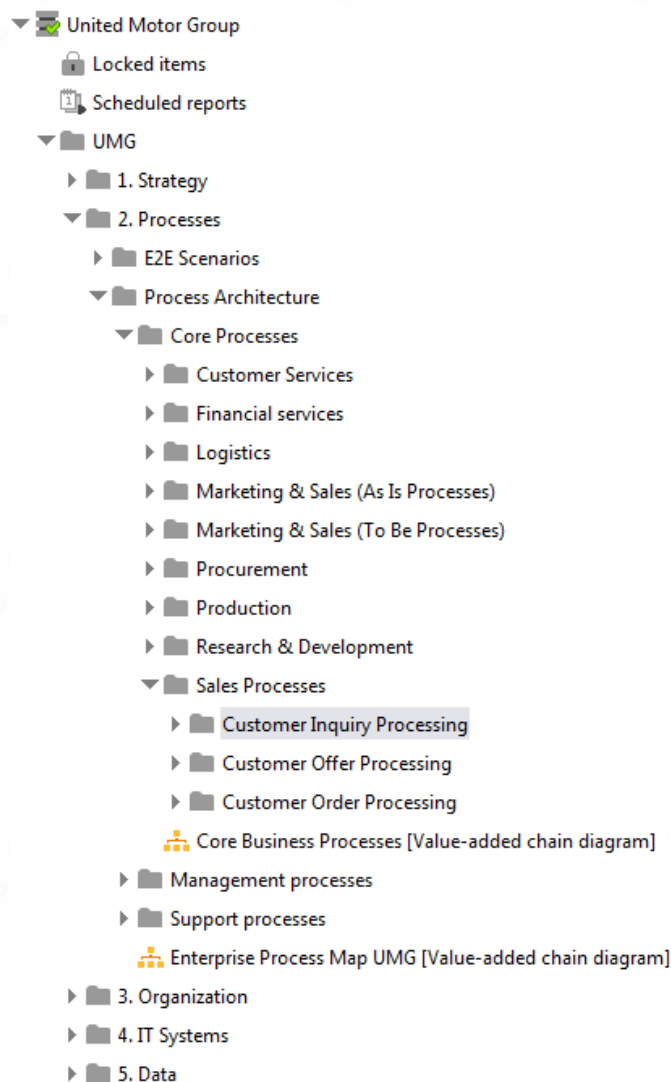
You want to expand the group structure of your database so that you will be able to navigate through the database as the number of models increase during the course of the pilot project.

The project team has decided to create process-independent groups (IT Systems and Data) for static model types. Models of the actual process documentation are to be stored in a process-oriented group structure (Core Processes).

In this exercise, you are to design a logical structure for an **ARIS** database.

Procedure

- 1) In the Explorer tab go to **United Motor Group** database.
- 2) Replicate the folder structure below:



Exercise 3 – Organizational Modeling

Map Organizational units or Positions to each process. As the first step, you want to map your company's organizational structure in your project database.

In this exercise, you will create a fragment and familiarize yourself with the **Organizational chart** model type.

Procedure

- 1) Create an **Organizational chart** and call it “**UMG Organizational Chart Europe**” within the group of your database entitled **3. Organization**.
- 2) Creating a fragment will allow you to model more quickly if the model has the same structure. Do this by creating the following model, utilizing the connection type is composed of.



- 3) Select the model you just created right-click and select **Create fragment**



Fragments are model building blocks that you can reuse. You can combine building blocks using all model items.

- 4) Name the fragment: **Fragment Ex3**
- 5) Model the **UMG Organizational Chart Europe** with the information supplied below:
 - a) The **UMG Europe Organizational unit** is composed of the following Organizational units:
 - Management Europe
 - Sales Europe
 - Development Europe
 - Production Europe
 - Marketing Europe
 - b) The **Sales Europe unit** is composed of the following Organizational units:
 - Direct Sales
 - Sales Management
 - Partner Sales

- c) The **Direct Sales Organizational unit** is composed of the following Organizational units:
- Direct Car Sales Eastern Europe
 - Direct Car Sales Western Europe
- d) **Direct Car Sales Western Europe** is composed of the following Organizational units:
- Sales Team Germany
 - Order Processing Germany
- e) The **Sales Team Manager Position** is organization manager for the **Sales Team Germany Organizational unit**. **Sales Team Germany Organizational unit** is composed of the following Positions:
- Sales Team Secretary
 - Sales Employee
- f) The **Order Team Manager Position** is organization manager for the **Order Processing Germany Organizational unit**. **Order Processing Germany Organizational unit** is composed of the following Positions:
- Order Handling Secretary
 - Order Handling Employee
- 6) Save and close the model.

Exercise 4 – Organizational Modeling / Attribute Editing

The existing Organizational chart is to be supplemented with further information. This information can be added in many ways.

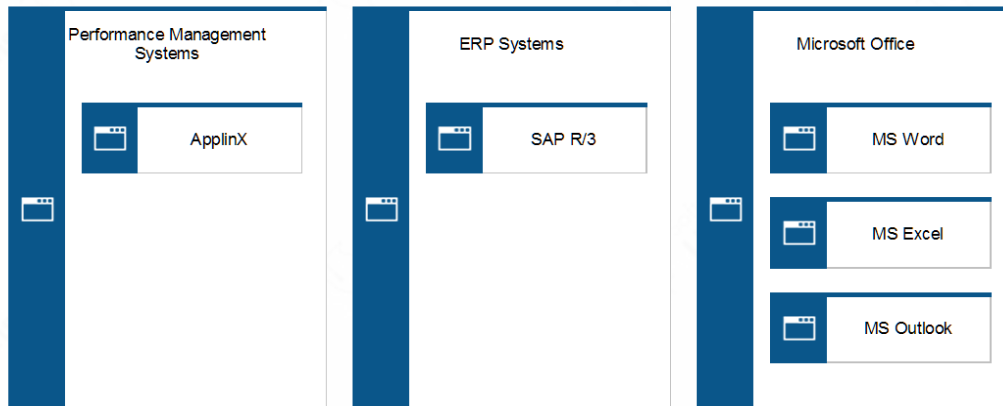
In this exercise, you are to learn how to expand existing models and maintain and place attributes.

Procedure

- 1) Open the **UMG Organizational Chart Europe** you created in Exercise 3.
- 2) Right-click on **Sales Employee**→**Attributes**→**System attributes**. The Link 1 attribute should link to **Job Description_Ex4.doc** (found in AEP_PDA_Materials/United Motor Group/04 Exercise). Enter the title as “Job Description” for your link in Title 1.
- 3) Next, place the Link 1 attribute to the **Sales Employee** Position. To do so, go to **Properties of Sales Employee**. Switch to the **Attribute placement**→**Add**. Check the “Show specified attributes” only checkbox and choose the Link 1 attribute, confirm with **OK**. Now place the attribute Link 1 in the lower right corner and confirm with **OK**.
- 4) Open the job description for the **Sales Employee** from ARIS by double clicking the object or **Right click**→**Go to**→**Link**.
- 5) Save and close the model.

Exercise 5 - Nested Objects in an IT Infrastructure Application System Type

The existing **Application System Type Diagram** needs to be supplemented with further information. The core team wants a clearer model and has set out its ideas in an illustration



(see figure below).

In this exercise you will learn to expand an existing model, become familiar with Nested Objects, familiarize yourself with the **Application System Type Diagram** model type and insert text from clipboard.

Procedure

- 1) Create an **Application System Type Diagram** and call it "IT Systems UMG" within the group of your database entitled 4. IT Systems.
- 2) Open the IT Systems_Ex5 Word document (found in AEP_PDA_Materials/United Motor Group/05 Exercise) - select and copy the list of applications. Return to **ARIS Architect & Designer**.
- 3) In the IT Systems UMG model right-click in the modeling area **Paste as→Object**.
- 4) Notice the text you copied is in the window. Select **Application system type** from the Symbol drop down menu then click **OK**.
- 5) Modify the copy so it meets the requirements of the core project team (pictured above) by utilizing Nested Objects (Object in Object).
- 6) Place the name of the "outer" object in the top center of the object as shown in the picture above.

- 7) Drag the three "outer" objects to an appropriate size, and place the smaller objects within the larger "outer" objects. The relevant implicit connection type is encompasses.
- 8) Save and close the model.

Exercise 6 – Occurrence and Definition Level

At the beginning of the project, you mapped your core business processes in a **Value-Added Chain Diagram (VACD)**. You saved this model in the **UMG** folder. During the course of the project you have fleshed out the database structure with details by creating subgroups.

Now you want to arrange the existing database content in a logical structure.

In this exercise, you are to understand the structure of **ARIS Explorer**, establish distinctions between Models and Objects or arrange Models and Objects in the **ARIS Explorer** structure

Procedure

- 1) Switch to the Explorer tab. Move the **Core Business Processes VACD** with its objects to the **UMG/2. Processes/Process Architecture/Core Processes** group.
- 2) Identify the group in which the Object definitions of the individual core business processes are located after you have moved the model.
- 3) Then move these Object definitions (**Research and Development, Sales, Production Development and Production Planning**) to the Explorer groups with the corresponding names.

Note

There are two very important reasons for storing Object definitions and Models in the "correct" Explorer groups:

- 1) Models and Objects can be located more easily.
- 2) Access privileges are issued with reference to Explorer groups in **ARIS**.

Exercise 7 – Create Occurrence Copy / Organizational Responsibility in VACD

At the start of the project you documented the core business processes of **Direct Car Sales** in a **Value-Added Chain Diagram (VACD)** in your database. In accordance with project conventions, you now want to map the organizational responsibility, in the form of Organizational units, in this process overview.

In this exercise, you are to add Organizational units to the high level process model (VACD) and learn to use occurrence copies.

Procedure

- 1) Open the Core Business Processes VACD which can be found in the UMG/2. Processes/Process Architecture/Core Processes group.
- 2) Integrate the organizational responsibilities into the model using the occurrence copies of Organizational unit's from **UMG Organizational Chart Europe** organizational chart created in Exercise 3.
- 3) Use the situation described below to integrate the organizational responsibilities:
 - a) The Organizational Unit **Management Europe** carries out **Research and Development**.
 - b) The Organizational Unit **Development Europe** carries out **Production Development**.
 - c) The Organizational Unit **Marketing Europe** carries out **Marketing**.
 - d) The Organizational Unit **Sales Europe** carries out both **Customer Care** and **Sales**.
 - e) The Organizational Unit **Production Europe** carries out **Production Planning**.



Make sure you use **ONLY** occurrence copies in order to avoid redundancies in the database. This is very important for later evaluation.

- 4) Select the **Management Europe** Organizational unit and go to **Properties** → **Occurrences** to navigate to the **UMG Organizational Chart Europe** Organizational Chart.
- 5) Save and close all models.

Exercise 8 – Incorrectly Event-driven Process Chain (EPC)

The following page shows an EPC that was created incorrectly. Point out the EPC modeling rules that it is breaking.

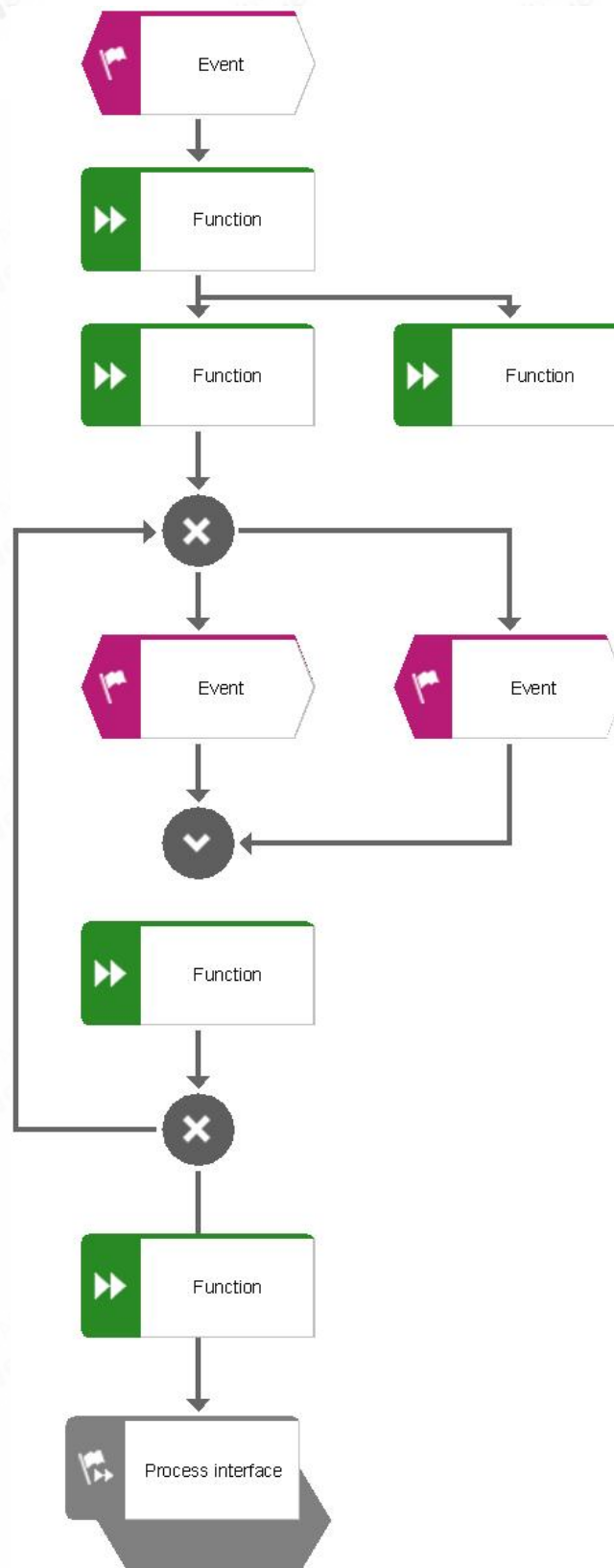
In this exercise, you are to familiarize yourself with the **Event-driven Process Chain (EPC)** and detect errors in modeling.

Procedure

- 1) On the following page is an incorrectly modeled EPC.
- 2) Check the structure of the model and point out the EPC modeling rules that this model is breaking – there are nine mistakes so take your time.

Note

There isn't "one" correct solution, various approaches are possible.



Exercise 9 – Process Overview with Even-Driven Process Chain (EPC)

You now want to break down the **Sales Function** of the **Core Business Processes** model using the EPC model type.

In this exercise, you will define a process flow with the EPC model, utilize the spell Check functionality to verify accuracy and apply EPC modeling rules and run layout generation.

Procedure

- 1) Create an EPC model **Sales Processes** in the **UMG/2.Processes/Process Architecture/Core Processes/Sales Processes** group. Then model it according to the situation below:
 - a) A **Customer Contact Exists** already, so create this Event to start the model.
 - b) The **Process Customer Contact Function** is initiated by this existing customer contact.
 - c) If the customer has shown an interest in the products/services as a part of customer contact process, a **Customer Inquiry Defined** Event should take place, which triggers the **Process Customer Inquiry Function**. Also, a **Customer Contact Not Successful** Event should be added as an alternative outcome for **Process Customer Contact**.
 - d) In case no **Customer Contact Exists**, a **New Customer Inquiry Received** can trigger **Process Customer Inquiry** as well.
 - e) Following **Process Customer Inquiry**, either an **Inquiry Registered** Event or if the required items are not available, an **Inquiry Not Accepted** Event occurs, thus terminating the process.
 - f) Next the **Process Customer Offer Function** starts. Spell “Customer” incorrectly to test the spell check functionality. Then either a **Customer Order Received** Event leads to the next Function or a **Customer Offer Rejected** Event takes place, which ends the process.
 - g) The **Customer Order Received** Event kicks off a **Process Customer Order Function**.
 - h) Lastly, either a **Customer Confirmation Sent** or **Order Data not OK** Event ends the process.
- 2) Switch to the **Format** tab → **Layout**, and define the following layout settings:
 - a) Minimum object spacing: Horizontal: “10”; Vertical: “10”;
 - b) Alignment: “Longest path-centered”;
 - c) Root node Position: “Arrange all roots at top”;
- 3) Apply the settings then save and close the **Sales Processes EPC**.

Exercise 10 – Assignments (existing model)

You have already modeled the **Core Business Processes** in a VACD. You also described the **Sales** aggregated Function detail with the **Sales Processes** EPC. Now you want to document in the database that the **Sales Processes** EPC describes the aggregated **Sales Function** in the **Core Business Processes** VACD.

In this exercise, you are to assign an existing model to an aggregated function.

Procedure

- 1) Open the **Core Business Processes** VACD which can be found in the **UMG/2. Processes/Process Architecture/Core Processes** group.
- 2) Right-click on the Function **Sales**→**New**→**Assignment** and assign the EPC **Sales Processes** which can be found in the **UMG/2. Processes/Process Architecture/Core Processes/Sales Process** group.
- 3) Save the changes to **Core Business Processes** VACD and close the model.

Exercise 11 – Occurrence Copy / Define Organizational Responsibility in EPC

Previously you have assigned the appropriate Organizational units to each Core Business Process as part of the **Core Business Processes (VACD)** model in your project database. You have also modeled the flow of processes for the **Sales Function** in the **Sales Processes EPC** at a general level. You now need to add Organizational units in the **Sales Processes EPC** based on the following information.

In this exercise, you are to expand an EPC with organizational elements and create occurrence copies.

Procedure

- 1) Open the **Sales Processes EPC** which can be found in the **UMG/2. Processes/Process Architecture/Core Processes/Sales Process** group. Add the responsible Organizational units listed in the situation below:
 - a) **Sales Team Germany** carries out the **Process Customer Contact Function**. All other functions in the process are handled by **Order Processing Germany**.



Make sure you use **ONLY** occurrence copies in order to avoid redundancies in the database. This is very important for later evaluation.

- 2) Save and close the model.

Exercise 12 – Detailed EPC / Assignments / Process Interfaces / Attribute Editing

Within the **Sales Processes** high-level EPC the **Process Customer Offer** Function must be further detailed with a vertical assignment to a new detailed EPC model. Going beyond the pure process flow, additional information (organizational responsibility, supporting IT systems and Cluster Data/ Model objects) will now be included. Organizational responsibility will be shown by the **Position** object, supporting IT systems are to be represented with the **Application System Type** object and **Cluster Data/ Model** objects for data elements.

In this exercise, you will learn to assign a detailed model to an aggregated function (assign new model to an object), use process interfaces and record times for multiple functions at once.

Procedure

- 1) Select the **Process Customer Offer** Function (object) which can be found in the **Sales Processes** EPC in the **UMG/ 2. Processes/Process Architecture/Core Processes/Sales Process** group.
- 2) Create an assignment for the **Process Customer Offer** Function. Save the new EPC model with the same name as the function, it is detailed in the **Customer Offer Processing Explorer** group.
- 3) Map the information supplied below in the **Process Customer Offer** EPC. Be sure to create occurrence copies of all objects already modeled in the database (the start and end Process Interfaces (Functions) and common Events of the superior **Sales Processes** EPC, organizational elements and Cluster data/ Model objects).
- 4) Follow the Event-Function-Event order (where no Event is given please create one).
 - a) Use the Function and Event(s) immediately before and after **Process Customer Offer** in the superior **Sales Processes** EPC as Process interface and common Event(s) to begin and end the new model.
 - b) **Order Handling Secretary** Position carries out **Copy Inquiry Data into Offer** Function using **MS Word Application** System Type and **Customer Inquiry, Customer Data** and **Customer Conditions** Cluster Data/Model objects with connection type is input for.
 - c) **Order Handling Employee** will **Check Availability of Article(s)** using the **SAP Application** System Type and the **Product Data** and the **Customer Inquiry** Cluster Data/Model objects using connection type is input for.
 - d) There is an XOR operator after this and a **Delivery NOT Possible** Event or a **Delivery Possible** Event.
 - e) After a **Delivery NOT Possible** Event, **Order Handling Employee** must **Agree on Alternative Delivery Date** with the customer - there is an XOR operator after this function.

- f) Then either a **Customer Offer Rejected** Event ends the process or a **Delivery Date Defined** Event allows the process to continue.
 - g) Use **XOR** to merge the open paths.
 - h) Next **Order Handling Employee** can **Reserve Article** in SAP using the **Product Data**.
 - i) After the articles are reserved, **Order Handling Secretary** will **Generate Offer** (function) in **MS Word**, the output of which is the **Customer Offer**, use has as output.
 - j) The Apprentice will then **Monitor Offer** (function) in **Applinx**.
 - k) The ending Event is **Customer Order Received** which leads to the final Function **Process Customer Order**.
- 5) Maintain the Average Processing Time for each of the Functions listed below:
- a) Highlight one function object then right-click on that function → **Select** → **Select all of this type** (this will have ARIS select all Function objects in your model).
 - b) Right-click on one of the selected functions → **Attributes**. On the left side select **Times** → **Processing time folder**. Complete the Average processing times for each function (see the table below).
 - c) Save the attributes.
 - d) Now, display the 'Average Processing Time' attribute in the top left corner (outside) of the Function and ensure that the data is displayed together with the 'Attribute Name'.

Function	Average Processing Time
Copy Inquiry Data into Offer	7 minutes
Check Availability of Articles	15 minutes
Agree on Alternative Delivery Date	4 minutes
Reserve Article(s)	2 minutes
Generate Offer	5 minutes
Monitor Offer	3 minutes

- 6) Save and close the **Process Customer Offer** model.

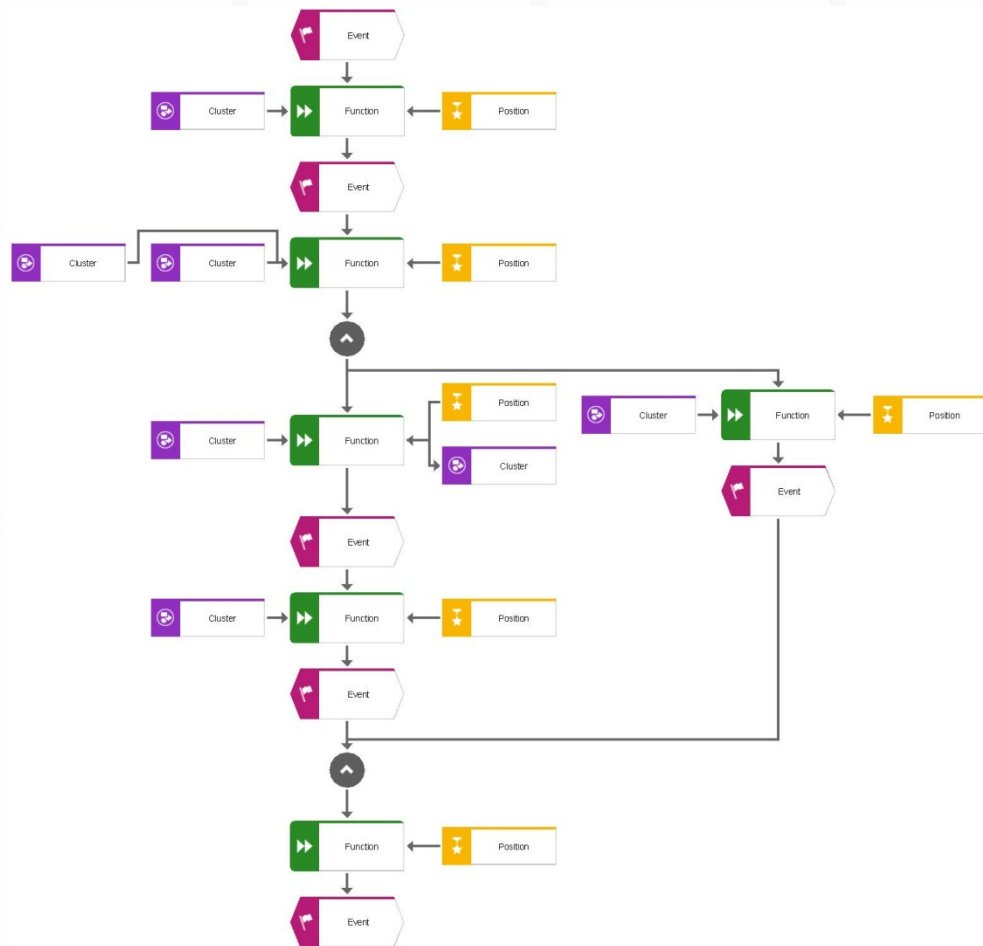
Exercise 13 – Detailed EPC / Assignments / Occurrence Copy

After **Process Customer Offer**, you want to model the previous process in the **Sales Processes EPC**, **Process Customer Inquiry**. Additional information about the supporting organizational responsibilities, data input and output, and IT systems support should also be modeled.

In this exercise, you will learn to describe an aggregated object in a new model, assign a new model to an object, use occurrence copies and use the Find Box.

Procedure

- 1) Assign a new EPC called **Process Customer Inquiry** to the **Process Customer Inquiry** Function (object) of the **Sales Processes EPC** which can be found in the in the **UMG/2. Processes/ Process Architecture/Core Processes / Sales Processes / Process Customer Inquiry** group.
- 2) Build the new **Process Customer Inquiry** EPC with the situation described below:
 - a) Use the Function and Event(s) immediately before and after **Process Customer Inquiry** from the superior **Sales Processes EPC** as Process interface and common event(s) to begin and end the new model.
 - b) To do so, copy the Function **Process Customer Contact** from the **Sales Processes EPC** which precedes the Inquiry function and also **Process Customer Offer** which is the Function that comes AFTER the Inquiry function. Ensure to copy the Events that immediately precede AND follow the Inquiry function with **Process Customer Offer** as well, as these will be the 'trigger(s)' and final results of the Inquiry process.
 - c) Create the EPC shown in the next page and select all objects. The connection type for the Cluster data/Model objects should be is input for and has output of and they are the main connection types for data. The connection type from the Position object to the Function should be carries out.
 - d) Start with the Function **Create New Customer Inquiry** which is carried out by the **Sales Team Secretary** with input of **Customer Inquiry**. Enter **Customer Inquiry Registered** as the Event after this function.
 - e) **Sales Employee** carries out **Configure Car** Function using **Product Data** and **Customer Inquiry** data (Clusters data/Model objects). The result is **Car Configured**.
 - f) Next **Sales Employee** carries out **Define Car Price** Function using **Product Data**. This price information is added to **Customer Inquiry** so this needs to be an input as well to this function. The result is **Car Price Defined**.



- h) After the result of **Discount Calculated** we have a parallel path, so don't forget to change the object selection to a rule, and then create the flow that shows the **Sales Employee** carrying out **Define Tax**, which uses **Product Data**, resulting in **Tax Defined**.
- i) These two parallel flows will now meet together to continue and you'll need the corresponding **Rule Object (AND)** to do this and then you can finish with the Smart design entry at this point.
- j) Once the taxes and the net car price have been calculated, from the concluding AND rule, a **Check Export Conditions** is performed by **Sales Employee** to determine whether an **Export is Allowed** or **Export is NOT Allowed**.
- k) If **Export is NOT Allowed**, the **Sales Team Secretary** must **Delete Customer Inquiry** with **Customer Inquiry** input, leading to the common Event (copied from the Sales Process) **Inquiry NOT Accepted**.

- l) If **Export** is **Allowed**, the **Sales Team Secretary** performs **Save Customer Inquiry** which has input/output as **Customer Inquiry**, leading to the **Inquiry Registered** common Event which continues to the **Process Customer Offer** Process Interface - both of which you copied from the **Sales Processes EPC** earlier.
- 3) Use the **Process Customer Offer** Process Interface to navigate from the **Process Customer Inquiry** EPC to the **Process Customer Offer** EPC and back again.
- 4) Save and close all models.

Exercise 14 – Modeling an Additional Detail EPC

After the **Process Customer Offer** and **Process Customer Inquiry** processes, you now want to model the fourth process step in the superior **Sales Processes EPC – Process Customer Order**. Only the organizational responsibility and supporting application software should be displayed with the control flow.

In this exercise, you are to increase your modeling skills and navigate horizontally within a modeling level using process interfaces.

Procedure

- 1) Assign a new EPC to the **Process Customer Order Function** (object) of the superior **Sales Processes EPC** which can be found in the **UMG / 2. Processes / Process Architecture/ Core Processes / Sales Processes** group.
- 2) Model the **Process Customer Order EPC** based on the following steps and save it in **UMG / 2. Processes / Process Architecture/ Core Processes / Sales Processes/ Process Customer Order** folder:
 - a) Use the Function and Event(s) immediately before and after **Process Customer Order** from the superior **Sales Processes EPC** as Process interface and common event(s) to begin and end the new model.
 - b) The function **Compare Offer Data with Order Data** takes place in **SAP**.
 - c) The result is either exclusively **Order Data is NOT OK** and this stops the process, or the **Order Data Matches Offer** Event and this leads to **Check Availability of Articles** carried out by the **Order Handling Employee** and the **SAP** system.
 - d) If an **All Articles Available** Event occurs, a **Reserve Articles** function is performed by the **Order Handling Employee** using the **SAP** system.
 - e) However, if the other exclusive result from the function above is **Not All Articles Available**, then the **Order Handling Employee** will **Create Production Order** in **MS Word** on a parallel path.
 - f) When all the articles have been reserved and/or committed for production, the **Order Handling Employee** does a **Confirm Order** in **MS Word**, ending the process with the **Customer Order Confirmation Sent** Event.
- 3) Use the **Process Customer Offer** process interface to navigate from the **Process Customer Offer EPC** to the **Process Customer Order EPC** and back again.
- 4) Save and close all models.

About

The **AEP - PD&A Architect Beginner Modeling Exercises- University Edition** is a part of a series of three tutorials:

- **AEP - PD&A Architect Beginner Modeling Exercises- University Edition**
- **AEP - PD&A Architect Advanced Modeling Exercises - University Edition**
- **AEP - PD&A Architect Administration Exercises - University Edition**

These tutorials are aimed to help students who do not have any specific knowledge of Process Model and Design and who would like to teach themselves how to design, implement and simulate a business process. Finishing all three tutorials will help students further improve their knowledge on Business Process Management, modeling and gain expertise in **ARIS Architect & Designer**.

We hope you enjoyed working with **ARIS Architect & Designer** and **AEP- PD&A Architect Beginner Modeling Exercises- University Edition**.

With this tutorial:

- You learnt the basics of Business Process Management
- You are now familiar with the various modules and functionality of **ARIS Architect & Designer**
- With the successive exercises you familiarized yourself with working on various model types- Value added chain diagrams (VACD), Organizational charts, Application System Type Diagram Event-Driven Process Chain (EPC)
- You are now comfortable with creating, testing and evaluating these models with the help of **ARIS Architect & Designer**

For more information on ARIS products, ARIS Education Packages and to network with Business Process Management experts, visit [ARIS Community](#).

Thank you!