USING INTEROPERABILITY STANDARDS EASE DESIGN AND AUTOMATED DEPLOYMENT OF BUSINESS PROCESSES LIKE CUSTOMER-SWITCHING

Cyril EFFANTIN
EDF R&D – France
cyril.effantin@edf.fr

Jean-Luc SANSON
EDF R&D – France
jean-luc.sanson@edf.fr

Sylvie MALLET
EDF R&D - France
sylvie.mallet@edf.fr

INTRODUCTION

Deregulation in European electricity market drives companies such as **EDF** (***E**lectricité **D**e **F**rance*) to open their IT systems in order to communicate with a growing number of external actors such as balance responsible parties, transmission system operators, grid operators, customers, suppliers…etc. However, these communications and data exchanges are difficult because of the lack of interoperability between IT Systems.
Actors involved in the opened market must exchange more and more, and that is why they are increasingly pushed to work through consortia in order to define business processes and share a common **UML** (***U**nified **M**odeling **L**anguage*) business process model. **ETSO** (***E**uropean **T**ransmission **S**ystem **O**perators*) and **ebIX** (*European forum for **E**nergy **B**usiness **I**nformation e**X**change*) are good examples of such consortia. However, it may be hard to ensure the respect of models into applications because of the lack of automation between the model and its implementations.

The question is how to improve UML business processes models thanks to standards in order to endorse those needs of interoperability and respect of business models between applications?
In 2005-2006, in a project called CIMERGY, EDF R&D worked on a prototype based on the **ebIX CuS** (***C**ustomer-**S**witching*), in order to get some answers to those previous concerns and needs.

CUSTOMER SWITCHING R&D PROTOTYPE

General presentation of the CuS Business process

The Customer Switching business process is dealing with the ability for customer to change and get a new supplier in the European electricity and gas market. In this use case, a customer may contact its new supplier who will send a "CuS request" to the relevant grid operator operating in the customer area. Then after having validated the data, the use case ends with a "CuS confirmation report" and the old supplier is notified of the change.

Model Driven Prototype approach

As EDF is very interested and active in the standardization, EDF R&D proposed to see how standards could help the design and deployment of applications based on such use cases.
And the main idea for this R&D prototype (see Figure 1) was to take advantage of standards in order to design the customer-switching process as a ***UML business model*** and then to ***monitor and drive interoperable applications implementations*** through Web Services orchestration architecture.

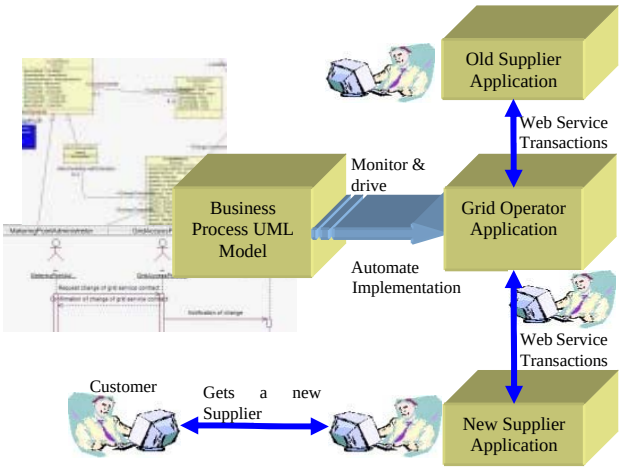

Figure 1: EDF R&D prototype for Customer Switching

Transactions based on Web Services layers imply that all business data integrated in those services are defined in **XML** (*e**X**tensible **M**arkup **L**anguage*).
Therefore, this prototype highlights another question: how to go from UML models to XML models? This document will show that we can automate the transformation of UML models into XML models thanks to the use of specific standards.

PRESENTATION OF MAIN CONSORTIA WORKS USED IN THE PROTOTYPE

UN/Cefact Core Component specifications

UN/Cefact (***U**nited **N**ations/Centre for Trade Facilitation and Electronic **B**usiness*) is the United Nations organism responsible for facilitating International Trade. With the arrival of Internet and XML technologies UN/Cefact started the **ebXML** (***E**-**B**usiness **XML***) project (jointly launched with OASIS organism) in order to improve **EDI** (***E**lectronic **D**ata **I**nterchange*). In particular, this project describes a technical approach in its **CCTS** work [3] (***C**ore **C**omponent **T**echnical **S**pecification*) in order to increase interoperability between applications exchanging XML messages.

CCTS embeds a methodology whose main idea is being able to:

- Share a technical way of designing XML messages schemas **XSD** (***X**ML **S**chema **D**efinition*).
- Share a common semantic (*called "Core Components"*) between XML messages. It is meant to be reusable inside messages exchanged though various business processes.

To achieve those requirements, UN/Cefact CCTS describes four different modeling levels (see Figure 2) that enable XML schemas building for message content which are based on a shared semantic defined into a UML information model.

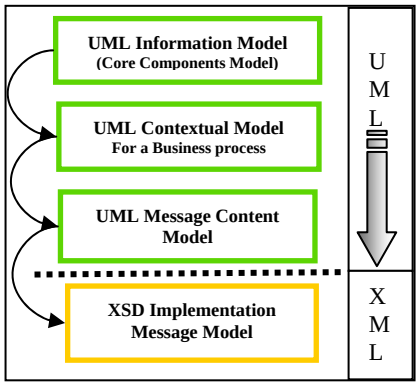

Figure 2 : UN/Cefact CCTS methodology

1. ***UML Information Model***: this UML model describes all the core concepts (Core Components) that could be found in a generic domain. This model will be shared by different Business Sub-Domains or business contexts.
2. ***UML Contextual Model***: in a second step, usually a business process, main core concepts of the information model are redefined (**BIE** ***B**usiness **I**nformation **E**ntities*) by adding restrictions on their properties in order to express business process specific needs. So new concepts cannot be added at this step because we can only restrict the information model. If we need core concepts that are not in the information model, the information model should be extended first.
3. ***UML Message Content Model***: the next step is to define what message architecture and content will be, or in another way how **BIEs** are assembled to create the Message.
4. ***Implementation Message Model***: for each UML message content model, an implementation message model could be produced in a specific syntax using an appropriate programming language. In our case, the syntax is XML, and we want to specify XML Schemas (XSD)

ETSO and ebIX

In 1999, ETSO (http://www.edi.etso-net.org/) was created as an association with ATSOI, UKTSOA, NORDEL and UCTE as founding association members. However, on June 2001 ETSO became an International Association. Today, the networks represented by ETSO supply more than 490 million people with electric energy. The consumption of electric energy amounts to approx. 3200 TWh per year. The length of HV (400 and 220 kV) lines covered by ETSO is of more than 290 000 km.

The Task Force 14 of ETSO focuses on EDI between all electricity market participants. Different national interchange standards mean that electricity suppliers for more than one country need heterogeneous IT solutions. Thus, the mission of ETSO Task Force 14 is to define and document EDI standards based on a general functional description of the electricity market in the European Union and on existing examples in the deregulated market. In order to reach its aim ETSO TF14 developed a Role Model which describes all the actors of the deregulated electricity market. The main issue of ETSO is the **ESS** (***E**nergy **S**cheduling **S**ystem*) document which is implemented in most of the member countries.

The purpose of ebIX (http://www.ebix.org) is to advance, develop and standardize the use of electronic information exchange in the energy industry. The focus is on interchanging administrative data for the internal European markets for electricity and gas. ebIX shall also cover the needs both for the wholesale market (upstream) and the retail market (downstream). ebIX will follow the rules of the European Union where applicable. As far as now ebIX has produced some very useful documents such as the Customer Switching business process study. ebIX consortium works closely with ETSO TF 14 and reuses the Role Model.

IEC TC57 / CIM UML model

IEC (*International Electrotechnical Commission*) TC57 (Power System Management and associated Information Exchange) is chartered with developing standards for electric power system management and associated information exchange in the areas of generation, transmission and distribution real-time operations and planning. The scope also includes information exchange to support wholesale energy market operations. See http://www.cimuser.org for more information.

The IEC TC57 working groups WG13 and WG14 are developing the UML information model called CIM (*Common Information Model*), which is the "core component" of the standard. The standard is composed of two parts: IEC 61970 and IEC 61968 [1]. IEC61970 defines common objects and attributes in UML, and a set of Common Services used by Energy Management Systems (EMS). IEC 61968, extends CIM model to Distribution and focuses on the definition of CIM-XML messages related to Distribution Business Process.

WHY CONSIDERING CIM, ETSO AND EBIX WORKS IN SUCH A PROTOTYPE ?

ETSO, ebIX – a large business working base to be considered in Europe

First, ETSO and ebIX are standardizing the way actors involved in the open energy market will exchange. Thus, their works represent a real relevant working base to see how it can match French specific needs. In 2004-2005, EDF R&D made a study to see if ebIX CuS UML model was relevant for EDF specificities. Even if EDF was not belonging to ebIX, at a higher level of modeling, ebIX UML business process models were quite matching EDF requirements.

So if ebIX is making a relevant work for the CuS business process, why should we consider as well the CIM UML model?

CIM UML model – a more generic semantic to leverage interoperable data integration between business processes

CIM UML model contains a more generic semantic than those involved in specific business processes such as the CuS. Indeed the CIM semantic endorses many other electricity domains. *As EDF is involved in many other business processes, EDF is interested in linking different semantics belonging to different business processes (ex: ebIX, ETSO…etc) with a more general one such as the CIM.*

Thus, how can we define the relationship between IEC TC57 and other consortia such as ebIX, ETSO?

At the moment, IEC TC57 is thinking about adopting UN/Cefact CCTS methodology in order to define XML messages based on the use of the *CIM semantic that could play the role of the UML information model in the CCTS* (see Figure 2). IEC would like also to integrate ETSO UML Business processes into the CIM standard. EDF R&D is participating in this integration and showed that the UN/Cefact approach could also be used to endorse the requirements of this integration.

Therefore, thanks to the use of those methodologies and standards, we can imagine that any utility such as EDF could exchange data with other external actors by following and respecting standardized business processes (like ETSO, ebIX, …etc) while sharing and forwarding those data into a CIM based semantic to internal IT system applications.

METHODOLOGY AND APPROACH USED IN THE PROTOTYPE

General approach

In order to automate the application code generation from the UML models, the prototype is based on a "**MDA**-like" approach (***M**odel **D**riven **A**rchitecture*) divided in three phases.

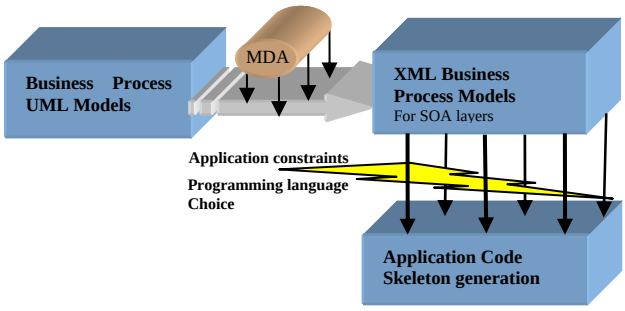

Figure 3 : General EDF R&D Prototype approach

Business requirements are defined in platform independent models (UML Models) and joined to platform dependent constraints dedicated to our specific Web Service architecture **SOA** layers (***S**ervice **O**riented **A**rchitecture*). It enables the creation of platform specific models which are in our case a set of XML files specifying our Business Process SOA layers. Then, the application code skeleton is automatically generated from those XML models after having set the relevant technical application constraints and chosen a programming language (Java in our prototype).

We are now going to give some more detail about each phase.

UML Business Process Model phase

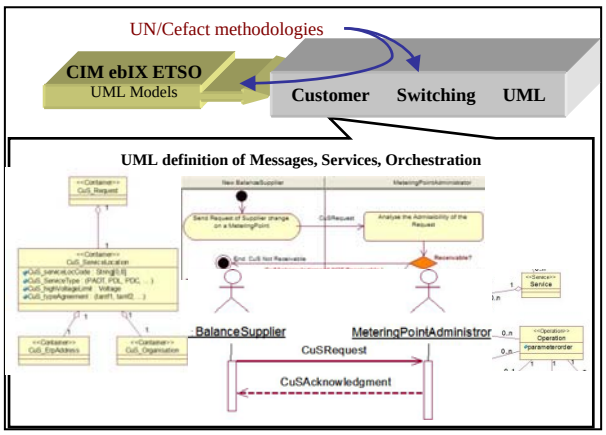

Figure 4 : UML Business Process Model Phase

When the UML study started, CuS ebIX UML models were not enough consistent and detailed compared to our prototype scope. So we only reused higher CuS ebIX UML views and the ETSO role model (extended by ebIX) linked to the CIM semantic thanks to the UN/Cefact CCTS methodology (see Figure 4). Then, those UML diagrams were completed by referring to EDF business knowledge.

This business process UML study was also influenced by **UN/Cefact UMM** (***U**nified **M**odeling **M**ethodology*). UMM is a Top-Down approach starting from the highest level use cases (ex: definition of main actors involved in the CuS use case). Then each use case is detailed and refined until the lowest modeling levels (generally the definition of each

interaction, message). In particular, each message is based on CCTS methodology applied to the CIM semantic. It lets us automate the generation (see Figure 2) of XML message model (XSD) that we need for the second phase.

XML Business Process Models phase

As the prototype was meant to be dedicated to SOA architecture, the UML business process model must be transformed automatically into Web Service oriented XML models (see Figure 5). In particular, we wanted the interaction knowledge to be separated from the treatments and computation knowledge. So we created a **BPMS** (***B**usiness **P**rocess **M**anagement **S**ystem*) containing the orchestration knowledge and managing each Web Service only dedicated to business computations and treatments.

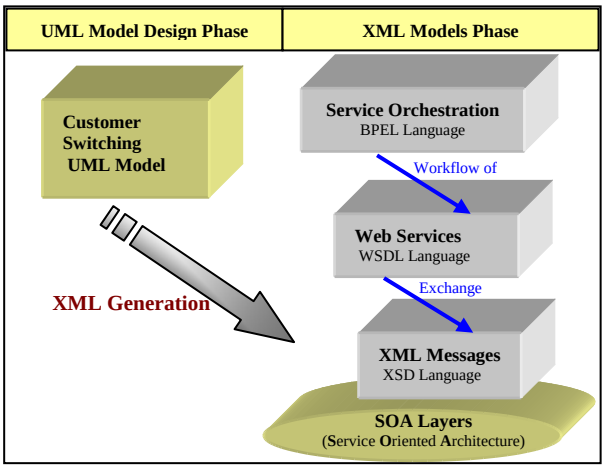

Figure 5 : XML Business Process Models Generation

In this phase, XML models are entirely based on standards such as **BPEL** (***B**usiness **P**rocess **E**xecution **L**anguage*) for the services orchestration, **WSDL** (*Web **S**ervice **D**escription **L**anguage*) for Web Services definition, and **XSD** for the definition of messages exchanged through those Web Services. Some of the XML files (BPEL, WSDL) were partially generated manually. Indeed, standards (ex: **BPMN** ***B**usiness **P**rocess **M**odeling **N**otation*) handling this automatic generation from UML models are too young and not enough UML tool supported.

Application code skeleton generation phase

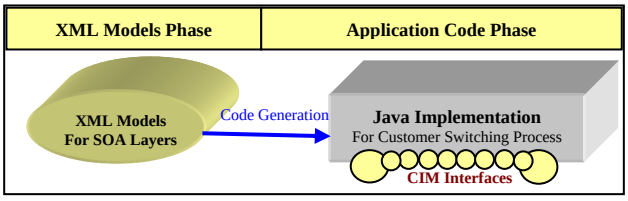

Figure 6 : Application code generation

The Java code skeleton for the prototype is generated automatically from the previous XML Business Process models (see Figure 6). In particular, the BPEL XML file can be directly imported into a BPEL engine producing, compiling, and deploying the relevant Java code for the workflow execution. This workflow is able to manage the interaction of each application component interface endorsed by Web Services WSDL. Java code skeleton generation for those interfaces is also automated from the WSDL XML files. Then, we just have to add the code for business treatments around each CIM messages transmitted by the Web Services.

CONCLUSIONS

The prototype highlighted that standards give different levels of facilitation for application deployment:

1. ***UML Business Process Design***:
   - To ease process design thanks to methodologies such as the UN/Cefact CCTS and UMM.
   - To share UML study between actors such as utilities.
2. ***Application deployment automation***:
   - To ensure the respect of those models inside applications thanks to the model driven approach reducing gap between business process experts and developers.
   - To build modular and evolutive applications.
3. ***Data integration and interoperability boost***:
   - To improve interoperability thanks to the use of a shared generic semantic (such as the CIM) deployed into messages thanks to UN/Cefact CCTS methodology.
   - To provide standardized Interfaces between applications.

The conclusions go over the scope of just the CuS use case and could be applied to many other business processes. For instance, some of the used standards are young and on the way of being more tool supported. However, this R&D approach appears as a proof of concept foreseeing what could be the future B2B (***B**usiness **T**o **B**usiness*) applications lowering integration costs of business processes.

ACKNOWLEDGMENTS

The authors would like to thank: Sylvie Anglade, Gilles Marius, Philippe Daguzan, Cyril Galabert, Eric Lambert, André Maizener, Dominique Voirin.
Greatest thanks also to IEC TC57 and CIM User Group members.

REFERENCES

[1] IEC 61970, 61968 documents, http://www.iec.ch
[2] E. Lambert, A. Maizener, F. Wu, J.L Sanson, C. Bahloul, P. Veyret, 2005, “Cimergy Project-First Results on IEC 61970-61968 standards EDF Implementation”, CIRED 2005, Turin.
[3] UN/CEFACT Core Components Technical Specifications, 2003, version 2.01