JAXFront @SwissRe

"Building a light-weight XML client based on the XML Rendering Engine JAXFront".

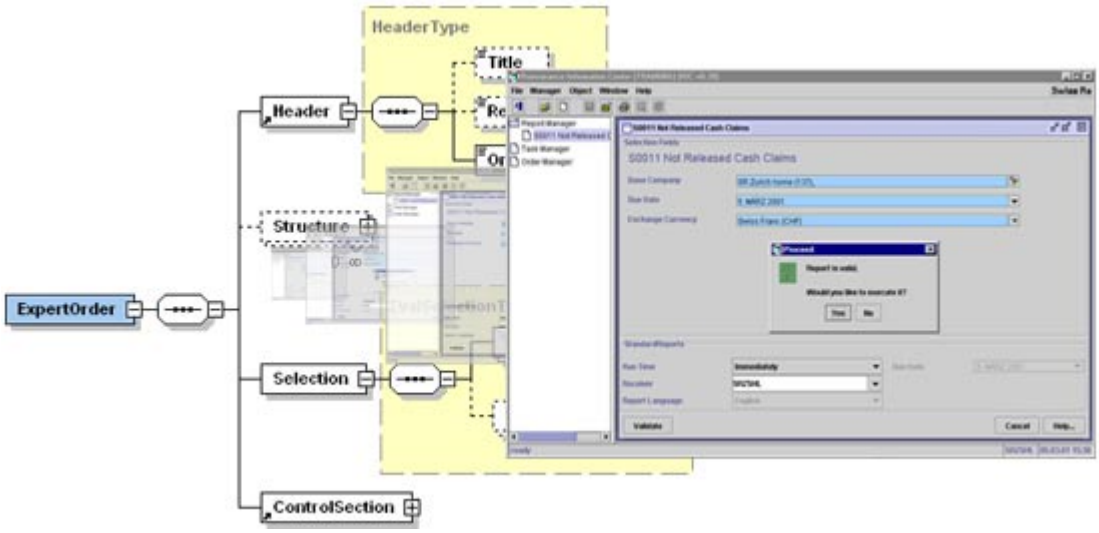
overview

SwissRe's Finance Division is responsible to provide customized reports for their own management, direct insurers, employees and agents. The number of reports which SwissRe is going to provide is growing every day. Each report needs a specific input mask and has own dependencies/restrictions between input fields (cross field rules). The goal was to let business analysts define their own input mask (report order) without touching or modifying the code and without the need to deploy the whole application again. Instead of creating static user interfaces for every type of report, a dynamic generation of the graphical front end has been realized with an XML rendering engine called JAXFront.

technical details

An asynchronous reporting system called RIC (Reinsurance Information System) has been built to provide all kinds of report data. SwissRe has built a CORBA Interface (IDL) to feed the RIC legacy system which is generating the raw report data through dynamic SQL. SwissRe together with *xcentric technology & consulting* defined an XML Schema which describes an abstract report order. This XML Schema is the only RIC server interface for the clients. The XML documents (streams) sent to the server will be transformed with XSLT to the CORBA IDL structure and forwarded to the RIC legacy system.

The input mask for a report order is generated dynamically on-the-fly on the basis of an XML Schema, an XML Template and a XUI Template. During the startup of the client, all available XML Templates (stored in a directory on the web server) will be listed in a tree view. The end user may choose now the report he wants to order. At this time, the rendering engine analysis the XML Schema structure and generates a type-safe graphical user interface in HTML or JavaSwing dynamically.



The validation of an input mask happens automatically based on the XML Schema data types, cardinalities, default values, etc. at client-side. Specific cross field rules and specific layout requirements are described in the XUI document. Every rule has an event (e.q. propertyChange), conditions and actions. The conditions are defined as valid XPath boolean expressions (e.q. /item/price * /item/quantity > 100).

technologies

Technology	*Usage*
XML	• Describing instances of report orders sent to the server. • Defining default templates for each standard report. • Defining UI look & feel (XUI).
XSLT	• Transform report orders to EBNF.
XPath	• Expressing event,conditions and actions in XUI. • Addressing XML Schema components in XUI.
XMLSchema	• Meta Modelling language for a report order.
Java JDK 1.2	• Implementation Language.
JAXFront 1.1	• XML Rendering Engine producing Java Swing GUI's.
CORBA OrbixWeb	• Middleware component.
DB/2 (XML Extender)	• Persistency of report orders.

Online Demo of JAXFront XML Rendering Engine: www.jaxfront.com

project benefits

- Rendering the XML orders means *no deployment* if a new report type is needed and provides a *common look and feel* (even on different UI channels like JavaSwing/HTML). The rendering engine will assure that only *valid XML streams* will be sent to the server. This results in a better data quality for all orders sent to the server and *reduces traffic/bandwith*.

- With the use of an XML Schema we were able to have a *common language* between the business & the IT through standardized, widely used modelling techniques. The XML Schema became the contract between the IT & the business.

- Due to a WYSIWYG XUI editor which allows to graphically define style and behaviour (rules) to define cross field rules, the business analysts are able to *realize their requests* on their own.

- The dependency on the IT team became less and less. The IT team could focus on really IT requests like infrastructure, system integration, interfaces, etc. instead of struggeling around with “GUI” issues.

- The whole development cycle could be reduced by 30 to 50%.

- SwissRe is able to launch new report types and make them available world wide within hours and almost without IT support.

contact information

xcentric IT Solution Architect:

xcentric technology & consulting
Mike Leber
Oberwiesenstrasse 5
CH-8057 Zürich

mike.leber@xcentric.ch
+41 (0)43 255 01 69

http://www.xcentric.ch
http://www.jaxfront.com

SwissRe IT Project Leader:

SwissRe
Oliver Masutti
Mythenquai 50/60
Postfach
CH-8022 Zürich

oliver_masutti@swissre.com
+41 (0)43 285 31 67

http://www.swissre.ch