GeLaBa – A Framework to Define Classes of XML Documents and to Automatically Derive Specialized Infrastructures

Benoît PIN / Georges-André SILBER
Centre de Recherche en Informatique / Mines de Paris

Why GeLaBa?

Abstract
GeLaBa means Générateur de Langage à Balises (in French), i.e. Markup Language Generator in English. It is a framework built upon a language called GML (GML stands for GeLaBa Markup Language) used to define classes of XML documents. Starting from a language-definition in GML, GeLaBa provides a collection of tools to automatically generate a complete specialized infrastructure to handle this specific class of XML documents.

Introduction
GeLaBa has been created when our research center began to participate to a project called LHEO funded by the french ministry of employment. The goal of LHEO was to create an XML schema to represent and exchange informations about professional formations in an uniform way in France. To suit the needs of all the users of this schema, several tools and GML became the core of GeLaBa. LHÉO is now an ongoing evolution of the language was very tricky. Because no existing tool suited our needs, we created a new language (GML) Based on XML and a set of generic tools to automate the creation and maintenance of LHÉO. Those tools and GML became the core of GeLaBa. LHÉO is now completely managed with GeLaBa, including its web site.

GML
The foundation of GeLaBa is GML, a class of XML documents that can describe a subset of all the classes of XML documents. It can represent fewer XML structures than the other schema languages (DTD, W3C XML Schema). It is also more regular, leading to simpler content models. This regular structure is sufficient in many cases and leads to simple but powerful tools. The two main constructions that we allow in GML are 1) an ordered sequence of unique elements that can be repeated several times (in DTD syntax, content of the form (a,b+)), and 2) an element chosen in a restricted set of elements (content of the form (a | b | c)*)

GML has features that are not present in any schema language. Elements and attributes can be typed with usual data types with various sizes (defined by the user). GML also supports the definition of dictionaries, whose keys can be used as enumerated types in elements or attributes. A dictionary is a simple list of entries, where each entry is a couple (key, value). Documentation is a fundamental aspect of GML: a multilingual documentation can be embedded in every component of the language definition (elements, attributes, dictionaries). Components can also have properties defined by the schema creator, to add extra informations not defined in the GML schema.

GeLaBa Tools

API
GeLaBa can automatically create an object oriented Application Programming Interface (API) with a specialized parser to read, modify and write documents in the defined language. The API validates the document and ensures that a modified document follows the schema before writing a file. This API is available under the form of Python classes, compatible with the Zope/Plinn platform. All classes are built upon SAX and DOM.

SCHEMAS
Usual schemas are derived from the schema represented in GML. DTD, W3C XML Schema, Relax NG. Those schemas have not the same power, but they can be used for several level of validations. For instance, the generated DTD can only express unbound repetitions whereas the generated W3C XML Schemas can express a maximal number of repetitions. The validation can also be done by our ad-hoc validator.

Documents
Documentations for the defined language are generated from the GML definition, in HTML and LaTeX (PDF).

GeLaBa tools

When to use GeLaBa?

Validator
An validator in Python is derived from the GML definition. This validator is more powerful and more strict than the generated schemas. It can use the dictionary services to validate dictionary keys contained in documents.

Examples generator
GeLaBa generates a complete solution on top of a complete Zope/Plinn application server to edit and store documents in the new language. It provides a persistent storage for the documents, under the form of an Object Oriented Database.

Why GeLaBa?

Resources

GeLaBa
http://www.gelaba.org

Subversion repository
http://svn.cri.ensmp.fr/trac/gelaba

Plinn
http://www.plinn.org

LHÉO
http://www.lheo.org

Thermodynamic DB Project (uses Plinn/GeLaBa)
http://www.ctdp.org

Acknowledgements

This work has been funded in part by DGEFP (French Ministry of Employment) for the LHÉO project. The authors would like to thanks DGEFP people (Jean-Michel Mazouth, Alain de Lorgeril, Stéphane Akonda) for their support. The authors would also like to thank Bertrand Blanchard from Items Media Concept for fruitful discussions about GeLaBa.