Model-Driven Engineering for implementing the ISO 19100 series of international standards

Cyril FAUCHER$^1$ and Jean-Yves LAFAYE$^2$

$^1$ IRISA/INRIA, Campus de Beaulieu, 35042, Rennes cedex, France
$^2$ Laboratoire L3i, Université de La Rochelle, 17042, La Rochelle cedex, France

cyril.faucher@irisa.fr
Outline

- Introduction and context
- Model-Driven Engineering and ISO standards
- Generate an end-user application by using the standards
- Conclusion
Introduction and context

- Manage ISO standards
  - many applications software implement ISO standards
  - but often these applications are black boxes

- Taking data providers requirements into account
  - more and more datasets
  - the types of these datasets is very heterogeneous
  - flexibility and genericity are required
Introduction and context

- **Why metamodelling**
  - a standard exchange is sufficient? Only to share data
  - data providers want to add new data structures without modify the software architecture
  - need for accessing the application structure to manage system
  - manage the elements building the application

- **Model-Driven Engineering (MDE)**
  - provide tools to manage applications at a high level of abstraction
  - decomposition in several modelling layers of the IS design
**Introduction and context**

- **ISO TC 211**
  - specification of the ISO 19100 series of standards

- **Several specifications**
  - data definition
  - data structure
  - metadata for georeferenced objects
  - aims at exhaustivity
  - provides embedded structures
Introduction and context

- **Contribution**
  - benefits of using MDE for prototyping and building an end-user application
  - the presented example exploits the various facets of the ISO 19100 series: metadata catalogue, the geographic features, the data structures and the data display
  - this work is part of an open-source project: Emios

- **EMIOS (Environmental Memory Interoperable Open Service)**
  - provide a range of services for storing and sharing information about environmental research activities
  - provide an extensible framework to facilitate the prototyping of IS dedicated to a study area
Outline

- Introduction and context

- Model-Driven Engineering and ISO standards

- Generate an end-user application by using the standards

- Conclusion
Model-Driven Engineering and ISO standards

- ISO standards integration into the MDE concept
  - paradigm that says everything is a model
  - several application standards of the series are based on this paradigm
  - another standard is designed, e.g.: ISO 19115 is defined by ISO 19110

A Feature Type is: « an abstraction of a real world phenomena » (from ISO specification)

From ISO 19109

From ISO 19110

From ISO 19115

From ISO 19117

Specific Feature Type for the definition of a Metadata model

GF_FeatureType

PF_FeaturePortrayal

FC_FeatureType

FC_FeatureCatalogue

Metadata Application Model

Dataset

conforms to
ISO 19117 metamodel defining a portrayal language

defines a language to specify the portrayal of FeatureTypes
Outline

- Introduction and context
- Model-Driven Engineering and ISO standards
- Generate an end-user application by using the standards
- Conclusion
Generate an end-user application by using the standards

- **Presentation of the example**
  - Implementation of a part of the Metadata standard model

- **Topcased graphical editor generator**
  - Open source solution
  - Templates for generating graphical editors
Generate an end-user application by using the standards

- Generation process of a graphical editor thanks to ISO 19100

1. ***.iso19110**
2. ***.iso19117**
3. ***.ecore**
4. ***.genmodel**
5. ***.diagramconfigurator**

- A portrayal specification related to the feature types
- Input resources
- 2 feature types has been defined
- Requirements defined by the IS administrator
- Generated models and used to generate the source code of the end-user application
- Generated Graphical Editor

**M D Metadata**
- entityId : EString
- data : EString
- contact : EString

**M D Identification**
- abstract : EString
- purpose : EString

**Generation multi-platform:**
Java, html, php
In our case, this is in Java (Topcased templates)
Generate an end-user application by using the standards

A specification example:
FeaturePortrayal: “identification1” linked to “MD_Identification” (instance of FeatureType)
  PortrayalRule “pr0” without link
    priority = 0 // default portrayal
    portrayalAction = (" MD_Identification.setFigureType("box") ") // figure type
    portrayalAction = (" MD_Identification.abstract.setFigureType("label") ")
    portrayalAction = (" MD_Identification.purpose.setFigureType("label") ")
    portrayalAction = (" MD_Identification.setColour(255,255,128) ") // yellow for the box
  background
  PortrayalRule “pr1” linked to “abstract” and “purpose”
    priority = 1
    queryStatement = (" if( abstract.getSize()==0 or purpose.getSize()==0 ) ")
    portrayalAction = (" MD_Identification.setColour(255,0,0) ") // red
  PortrayalRule “pr2” linked to “abstract”
    priority = 1
    queryStatement = (" if( abstract.getSize()>25 ) ")
    portrayalAction = (" abstract.subString(25,0) ") // subString method

External functions that are used:
getSize(String text) : int
setFigureType(String type)
setColour(int R, int G, int B)
subString(int size, int start) : String
PortrayalCatalogue: “PC1” contains {identification1, getSize, setFigureType…}
Generate an end-user application by using the standards

- Generated end-user application

  - Graphical editor
  - Well-formed model: model checking ability is provided *de facto*
  - Saving in an interoperable way (xmi-xml)

  Metadata Model Outline
Conclusion

- Interest of Model-Driven Engineering (MDE) for implementing part of the ISO standards
- Fast prototyping abilities to design an Environmental IS
- Generic feature of the approach to solve specific issues
Any question?

http://emios.sourceforge.net