

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

Cyril FAUCHER<sub>1</sub> and Jean-Yves LAFAYE<sub>2</sub>

<sub>1</sub> IRISA/INRIA, Campus de Beaulieu, 35042, Rennes cedex, France

<sub>2</sub> Laboratoire L3i, Université de La Rochelle, 17042, La Rochelle cedex, France

[cyril.faucher@irisa.fr](mailto: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





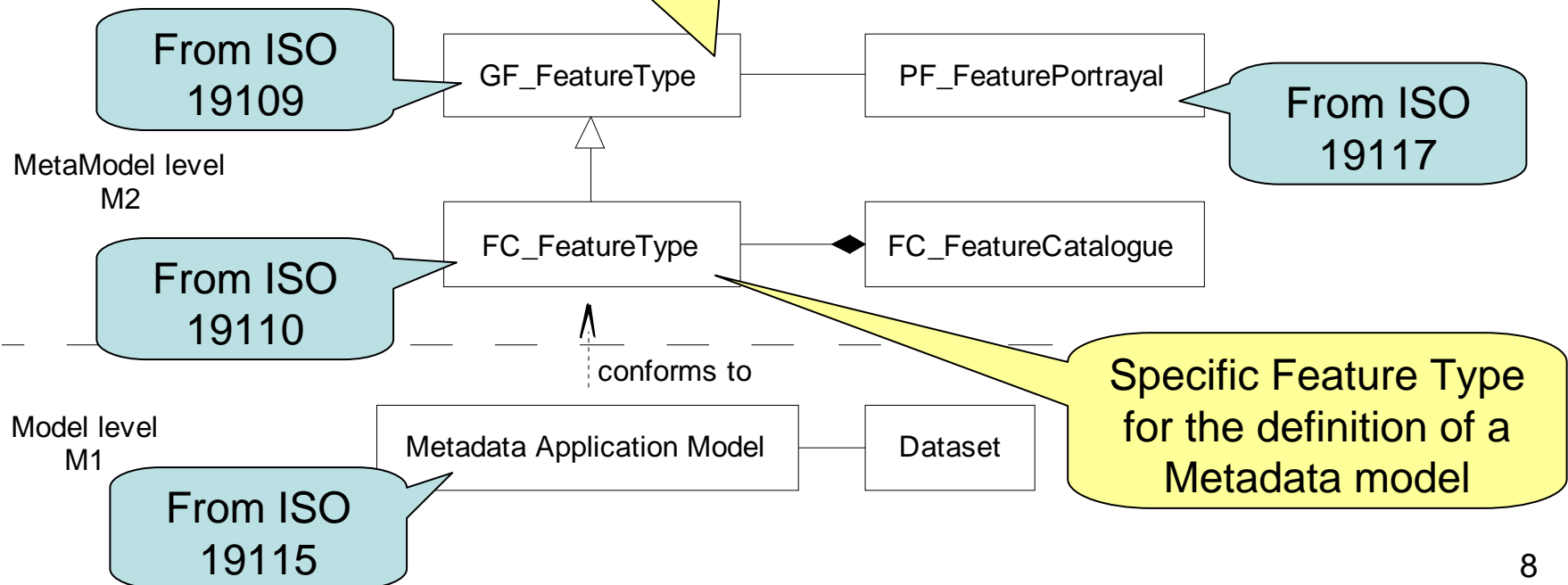
## ***Outline***

- 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

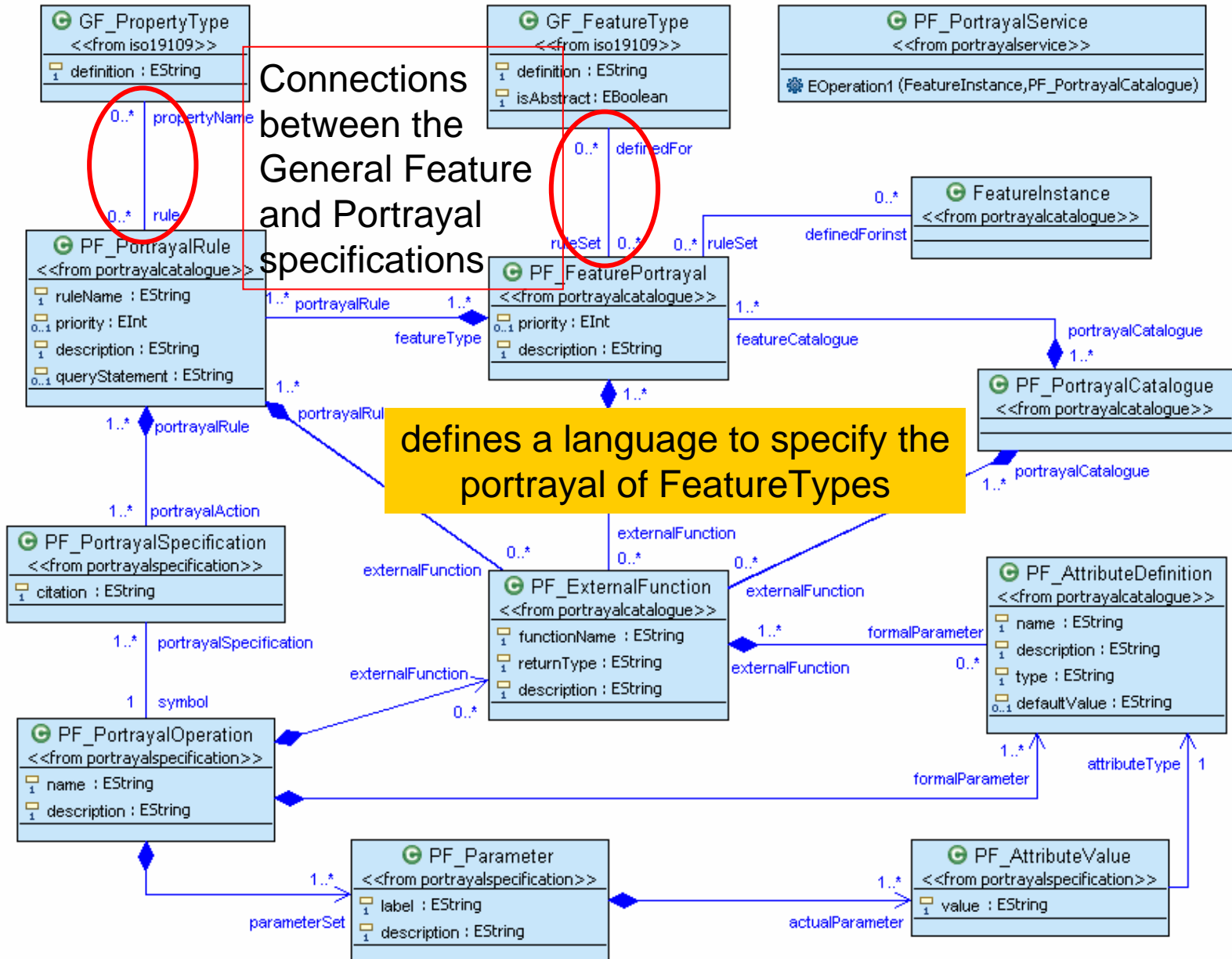
## ISO standards integration into the MDE concept

- paradigm that says everything is a model
- A Feature Type is: « an abstraction of a real world phenomena » (from ISO specification)
- other standard is designed, e.g.: ISO 19115 is defined by ISO 19110





# ISO 19117 metamodel defining a portrayal language





## ***Outline***

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

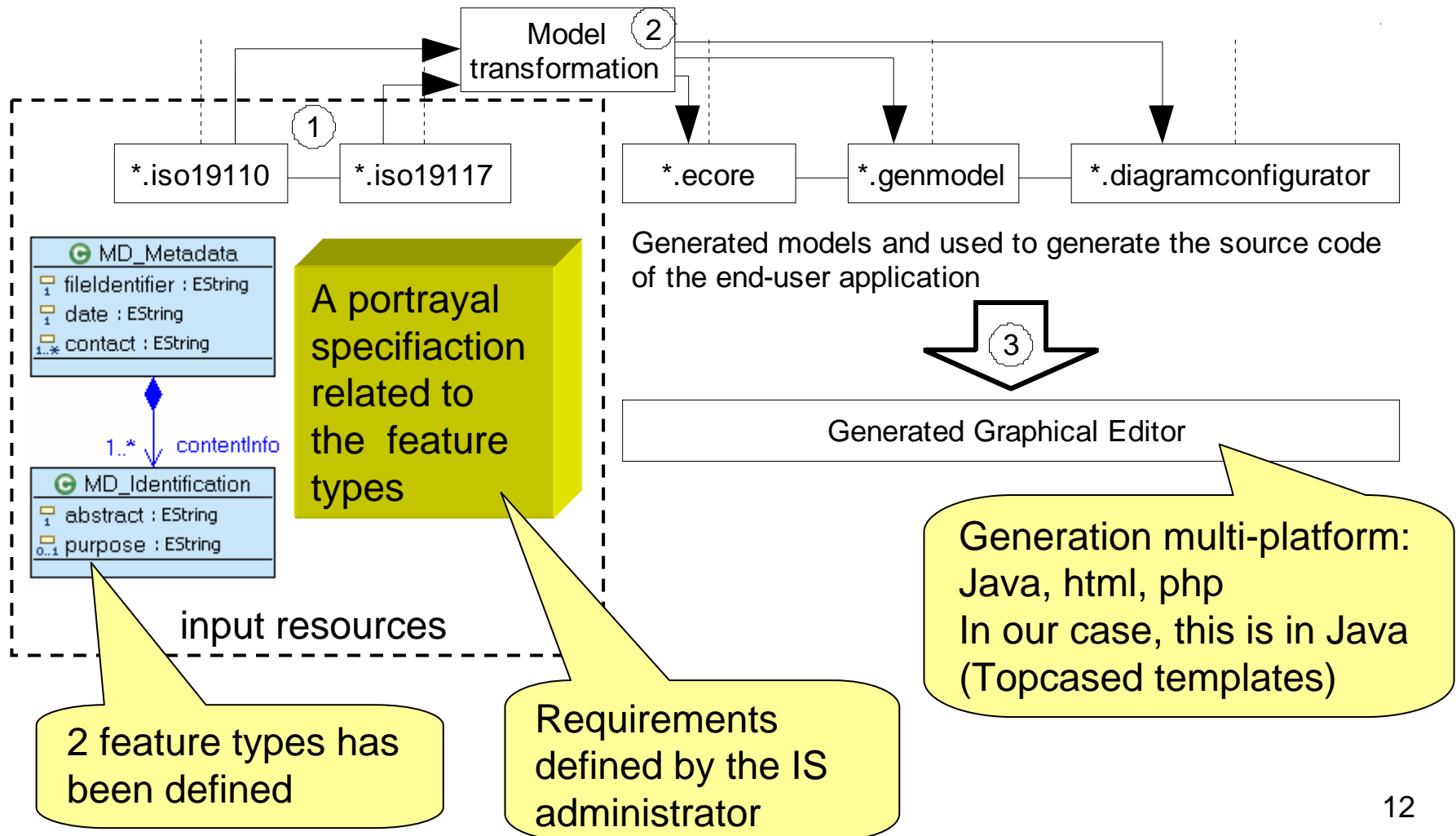


## ***Generate an end-user application by using the standards***

- **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



# 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

The screenshot displays the Eclipse IDE interface for Topcased Modeling. The main workspace shows a graphical editor for an ISO 19115 metadata model. The left sidebar contains a Metadata Model Outline, and the bottom right shows a Property view for the selected metadata instance.

**Graphical editor**

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

**Metadata Model Outline**

**Property view**

Property	Value
Abstract	This metadata describes the CoastGIS example
Purpose	The purpose of this metadata is to show the result of our work



## ***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>

