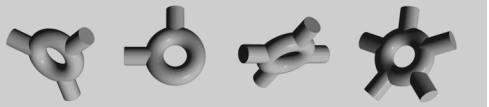
elements



Languages Package

TARMS Inc.

September 07, 2000

Typeset in LaTeX.

Contents

1 Interfaces **2**
- 1.1 Language . . . 2
  - 1.1.1 Relationships . . . 3
  - 1.1.2 Operations . . . 3

2 Classes **4**
- 2.1 LanguageModel . . . 4
  - 2.1.1 Relationships . . . 4
  - 2.1.2 Attributes . . . 4
  - 2.1.3 Operations . . . 4
- 2.2 LanguageReferenceDataModel . . . 5
  - 2.2.1 Relationships . . . 5
  - 2.2.2 Operations . . . 5

3 Associations **6**
- 3.1 language . . . 6
- 3.2 business language . . . 6
- 3.3 languages . . . 6
- 3.4 model . . . 7
- 3.5 parent . . . 7

4 Extensions to the Locations Package **9**
- 4.1 Location . . . 9
  - 4.1.1 Relationships . . . 9
  - 4.1.2 Operations . . . 9
- 4.2 LocationModel . . . 9
  - 4.2.1 Relationships . . . 9
  - 4.2.2 Attributes . . . 9

5 Extensions to the Dates Package **10**
- 5.1 DateFormat . . . 10
  - 5.1.1 Relationships . . . 10
  - 5.1.2 Operations . . . 10

List of Figures

1 Class Diagram— Languages . . . 8

List of Tables

1 Languages— Associations 6

Package Description

The Languages package provides basic descriptions of the languages used throughout the world. It allows the handling of language information for such things as date formatting.
The Languages model follows an inheritance pattern that allows languages to hold a parent language to accomodate the notion of language dialects.

The Languages Package extends:
• The Dates Package with information on the language that the date format uses.
• The Locations Package with information about the languages used in a location.

1 Interfaces

1.1 Language

The main use of the language interface is to provide a common point for information that is likely to vary on a per-language basis. For example, date formats, address formats, salutations.

Languages are assumed to form a hierarchy, with dialects and variants inheriting from the parent languages. For example, English has many variants, including US English, UK English and Australian English. Creoles, pidgins and other multiple-parent languages are not covered by this model.

1.1.1 Relationships

	Class	Description	Notes
⇑	Identifiable		
↑	Validatable		
↓	LanguageModel §2.1		
↓	LanguageReferenceDataModel §2.2		
↔	LocationModel	business language 0..n	
↔	LocationModel	languages 0..n	
↔	LanguageReferenceDataModel §2.2	model	
↔	LanguageModel §2.1		→
↔	LanguageModel §2.1	parent 0..1	

⇑:Inherits ↑:Realizes ↓:Realized by ↔:Association →:Navigable ◊:Aggregate ♦:Composite

1.1.2 Operations

Language parent() parent

The parent language, if this language is a dialect.

If the language is a dialect, this will return the language the dialect derives from. If the language is not a dialect, this will return nil.

StandardizedIdentifier code() code

The identifier for this language.

Returns the standardized identifier for this language. This identifier is (usually) the ISO 639 code.[1]

String name() name

The name of the language, in the common language for this system.

Returns the name of this language in the common language that this system uses. For example, if this system uses English as a common language, then the name will return *French* for the French language.

String localName() localName

The name for this language in this language.

Returns the name of this language in the language itself. For example, the French language is *Français* in French.

DateFormat dateFormat() dateFormat

The date format for this language.

Returns the date format that this language normally uses.

2 Classes

2.1 LanguageModel

A concrete implementation of the Language interface, using attributes to hold the basic information about the language.

2.1.1 Relationships

	Class	Description	Notes
↑	Language §1.1		
↔	DateFormatModel	language 0..n	
↔	LanguageReferenceDataModel §2.2		→
↔	Language §1.1		
↔	Language §1.1	parent 0..1	→

↑:Realizes ↔:Association →:Navigable ◊:Aggregate ⧫:Composite

2.1.2 Attributes

code: StandardizedIdentifier The standardized identifier code for this language.

name: String The name of this language in the common system language.

localName: String The name of the language in the language itself.

identifier: String The unique identifier for this language.

2.1.3 Operations

Reportable validate() validate

This method returns an object of type Reportable which will contain all the errors and warnings generated from this method. Below is a list of the axioms that must hold for an instance of this class to be valid. Each statement is followed by the error or warning message that will be issued if the axiom is violated.

- Both the name and the localName attributes cannot be nil.
- The name attribute must be unique.
- If the parent is not nil, the parent Language must exist.

- If there is no associated date format, then there must be a parent language and that language must have a date format.
- None of the languages ancestors or descendants can be itself. Languages must form a hierarchy and cannot form a loop.

«Static Method» Language standardLanguage() standardLanguage

The default language for this system.

Returns the language that is common throughout the system.

2.2 LanguageReferenceDataModel

A wrapper for the LanguageModel class which can be used to manage languages as part of reference data. This class implements the Language interface by delegating to the associated model.

2.2.1 Relationships

	Class	Description	Notes
⇑	ReferenceDataModel		
↑	Language §1.1		
↔	Language §1.1	model	→
↔	LanguageModel §2.1		

⇑:Inherits ↑:Realizes ↔:Association →:Navigable ◊:Aggregate ♦:Composite

2.2.2 Operations

String defaultDescription() defaultDescription

Default description of data.

Returns the name returned from the associated model.

Reportable validate() validate

This method returns an object of type Reportable which will contain all the errors and warnings generated from this method. Below is a list of the axioms that must hold for an instance of this class to be valid. Each statement is followed by the error or warning message that will be issued if the axiom is violated.

Returns the validation results for the superclass, composed with the validation results from the associated model.

3 Associations

Table 1: Languages— Associations

Association Role	Class	Card.	Notes
language			
language	LanguageModel §2.1	1	→
date format	DateFormatModel	0..n	
business language			
business language	Language §1.1	0..1	→
location	LocationModel	0..n	
languages			
languages	Language §1.1	0..n	→
location	LocationModel	0..n	
model			
model	Language §1.1		→
reference data	LanguageReferenceDataModel §2.2		
parent			
	Language §1.1	0..1	→
	LanguageModel §2.1	0..1	

→:Navigable ◊:Aggregate ♦:Composite

3.1 language

Role: language *Navigable* LanguageModel, 1.
Role: date format DateFormatModel, 0..n.
The language that the date format uses.

3.2 business language

Role: business language *Navigable* Language, 0..1.
Role: location LocationModel, 0..n.
The business language for this location.

3.3 languages

Role: languages *Navigable* Language, 0..n.
Role: location LocationModel, 0..n.
The languages in common use in this location.

3.4 model

Role: model *Navigable* Language.
Role: reference data LanguageReferenceDataModel.

3.5 parent

Role: *Navigable* Language, 0..1.
Role: LanguageModel, 0..1.

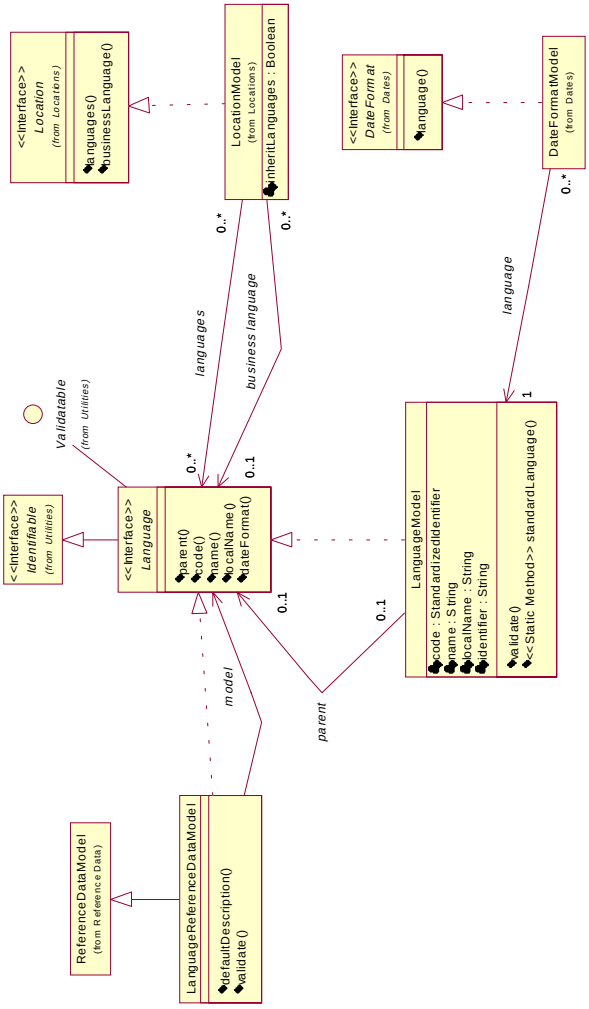


Figure 1: Class Diagram— Languages

4 Extensions to the Locations Package

4.1 Location

4.1.1 Relationships

	Class	Description	Notes
↓	LocationModel		

↓:Realized by

4.1.2 Operations

Set<Language> languages() languages

The location's languages.

Returns the set of languages that this location recognizes. Generally, the language set for this location consists of the languages of the parent location along with any local languages.

Language businessLanguage() businessLanguage

The most common business language of this location.

Returns the language most commonly used for business agreements in this location. If there is no specific business language for this location, the business language of the parent location is inherited.

4.2 LocationModel

4.2.1 Relationships

	Class	Description	Notes
↑	Location		
↔	Language §1.1	business language 0..1	→
↔	Language §1.1	languages 0..n	→

↑:Realizes ↔:Association →:Navigable ◊:Aggregate ♦:Composite

4.2.2 Attributes

inheritLanguages: Boolean Inherit languages from the parent location ?

5 Extensions to the Dates Package

5.1 DateFormat

5.1.1 Relationships

	Class	Description	Notes
↓	DateFormatModel		

↓:Realized by

5.1.2 Operations

Language language() language

The language that the date format uses.

Return the language that the date format uses for month and day of week names, etc.

References

[1] International Organization for Standardization (ISO). *Code for the Representation of Names of Languages*, number ISO 639, 1988. http://www.iso.ch/cate/d4766.html.