

Languages Package

TARMS Inc.

September 07, 2000

TARMS Inc.

http://www.tarms.com

Copyright ©2000 TARMS Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this model and associated documentation files (the "Model"), to deal in the Model without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Model, and to permit persons to whom the Model is furnished to do so, subject to the following conditions:

- The origin of this model must not be misrepresented; you must not claim that you wrote the original model. If you use this Model in a product, an acknowledgment in the product documentation would be appreciated but is not required. Similarly notification of this Model's use in a product would be appreciated but is not required.
- 2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
- 3. This notice, including the above copyright notice shall be included in all copies or substantial portions of the Model.

THE MODEL IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MODEL OR THE USE OR OTHER DEALINGS IN THE MODEL.

Typeset in LATEX.

Contents

1	Inte	faces	2
	1.1	Language	2
		1.1.1 Relationships	3
		1.1.2 Operations	3
2	Clas	es	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	Asso	iations	6
	3.1	language	6
	3.2	business language	6
	3.3	languages	6
	3.4	model	7
	3.5	parent	7
4	Exte	usions 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	Exte	usions to the Dates Package	10
-	5.1		10
		5.1.1 Relationships	10
		5.1.2 Operations	10
			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	Re	ations	ships
-------	----	--------	-------

	Class	Description	Notes
↑	Identifiable		
\uparrow	Validatable		
\downarrow	LanguageModel §2.1		
\downarrow	LanguageReferenceDataModel §2.2		
\leftrightarrow	LocationModel	business lan- guage 0n	
\leftrightarrow	LocationModel	languages 0n	
\leftrightarrow	LanguageReferenceDataModel §2.2	model	
\leftrightarrow	LanguageModel §2.1		\rightarrow
\leftrightarrow	LanguageModel §2.1	parent 01	

 $\uparrow:$ Inherits $\uparrow:$ Realizes $\downarrow:$ Realized by $\leftrightarrow:$ Association $\rightarrow:$ Navigable $\Diamond:$ Aggregate $\blacklozenge:$ Composite

1.1.2 Operations

Language 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()

The identifier for this language.

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

String 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()

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()

The date format for this language.

parent

code

name

dateFormat

localName

3

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				
\uparrow	Language §1.1						
\leftrightarrow	DateFormatModel	language 0n					
\leftrightarrow	LanguageReferenceDataModel §	2.2	\rightarrow				
\leftrightarrow	Language §1.1						
\leftrightarrow	Language §1.1	parent 01	\rightarrow				
	$\uparrow: Realizes \leftrightarrow: Association \qquad \rightarrow: Navigable \ \Diamond: Aggregate \ \blacklozenge: Composite$						

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()

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.

validate

- 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()	standardLan-
The default language for this system.	guage
Detume the lenguage that is common throughout the 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		
\leftrightarrow Language §1.1	model	\rightarrow
\leftrightarrow LanguageModel §2.1		
	→:Navigable ◊:Aggregate ♦	:Composite

2.2.2 Operations

String defaultDescription()

Default description of data. Returns the name returned from the associated model.

Reportable 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.

defaultDescription

validate

3 Associations

Ta	able 1: Languages— Associations		
Association			
Role	Class	Card.	Notes
language			
language	LanguageModel §2.1	1	\rightarrow
date format	DateFormatModel	0n	
business language			
business language	Language §1.1	01	\rightarrow
location	LocationModel	0n	
languages			
languages	Language §1.1	0n	\rightarrow
location	LocationModel	0n	
model			
model	Language §1.1		\rightarrow
reference data	LanguageReferenceDataModel §2.2		
parent			
	Language §1.1	01	\rightarrow
	LanguageModel §2.1	01	

 \rightarrow :Navigable \diamond :Aggregate \blacklozenge :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.

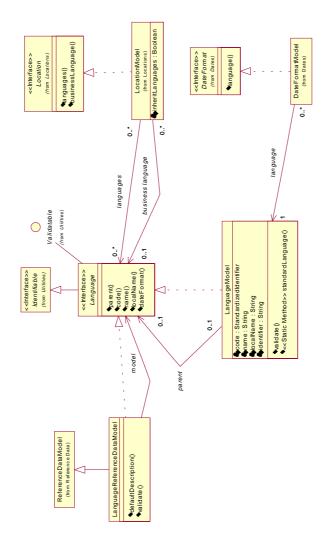


Figure 1: Class Diagram— Languages

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()

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()

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		
\leftrightarrow Language §1.1	business lan- guage 01	\rightarrow
\leftrightarrow Language §1.1	languages 0n	\rightarrow
↑:Realizes ↔:Association	→:Navigable ◊:Aggregate ♦:Co	omposite

4.2.2 Attributes

inheritLanguages: Boolean Inherit languages from the parent location ?

languages

businessLanguage

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()

The language that the date format uses.

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

language

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.