Generated File

# SIMFProfile

## Diagram: SIMF Profile

423988531.emf

1. SIMF Profile

# SIMFProfile::SIMF Computation Rules

Computation rules define mappings that are implemented via external methods. As such the implementation is defined by implementations, not the specification.

## Diagram: SIMF Computation Rules

1940672626.emf

1. SIMF Computation Rules

## Class ExistsRule

<<Exists Rule> is a rule to map the existence of an <element> to a boolean.<exists> is true iff <element> is not null.

### Direct Supertypes

[Rule Computation](#_7b2ad80ba8e04ff9ff1cb7b89d9ea1d3)

**package** SIMF Computation Rules

### Attributes

-1258462637.emf exists : [Boolean](#_6119a00b0834641b9fe3f5ae9f58237f)

Boolean that will be true if <element> is not null.

-1258462637.emf element

If <element> in not null, <exists> will be true.

## Class List First

The <List First> rules will take the <list> property and place the first element into<first>. If <list> is empty, <first> will be empty.If there are more <list> elements than 1, all remaining elements are placed as a set in <remainder>.If <list> is an un-ordered set the order will be indeterminate but repeatable.<<List First>> is bidirectional and will compute <list> by appending <first> and <remainder>.Note that this will act like a LISP CDR/CAR pair

### Direct Supertypes

[Rule Computation](#_7b2ad80ba8e04ff9ff1cb7b89d9ea1d3)

**package** SIMF Computation Rules

### Attributes

-1258462637.emf list [\*]

-1258462637.emf first [0..1]

-1258462637.emf remainder [\*]

## Class MapID

<<MapID>> is a rule where the source is an ID and the target is a class, maps an instance of the ID to an instance of the class.

### Direct Supertypes

[Rule Computation](#_7b2ad80ba8e04ff9ff1cb7b89d9ea1d3)

**package** SIMF Computation Rules

### Attributes

-1258462637.emf id : [Classifier](#_5f9a87915e1e9718a1a1cc45af995a70)

The source type (ID)

-1258462637.emf identified : [Classifier](#_5f9a87915e1e9718a1a1cc45af995a70)

The identified type (Identified by ID)

## Class Rule Computation

<<Rule Computation>> is an abstract supertype for a facade that includes external implementation. The implementation is outside of this specification.

**package** SIMF Computation Rules

## Class Summarize

<<Summarize>> is a rule that produces a natural language description of an element. Summarize may not be bi-directional and is expected to have information loss.<summary> is a summary of <element>. Content of summary is implementation specific.

### Direct Supertypes

[Rule Computation](#_7b2ad80ba8e04ff9ff1cb7b89d9ea1d3)

**package** SIMF Computation Rules

### Attributes

-1258462637.emf summary : [String](#_e8a6ce315d976318da3ab784a645ea44)

The element summary

-1258462637.emf element

The summarized element.

# SIMFProfile::SIMF Rules Profile

The SIMF rules profile defines the way to model rules and mapping within and between data sources via a conceptual model.

## Diagram: SIMF Rules Profile

603307669.emf

1. SIMF Rules Profile

Computation computes a value for the mapping end based on the expression applied to the mapped property or relationship. Where computation is used inverse mapping is not specified - any inverse mapping is implementation specific.

# Tree

## Class Hierarchy