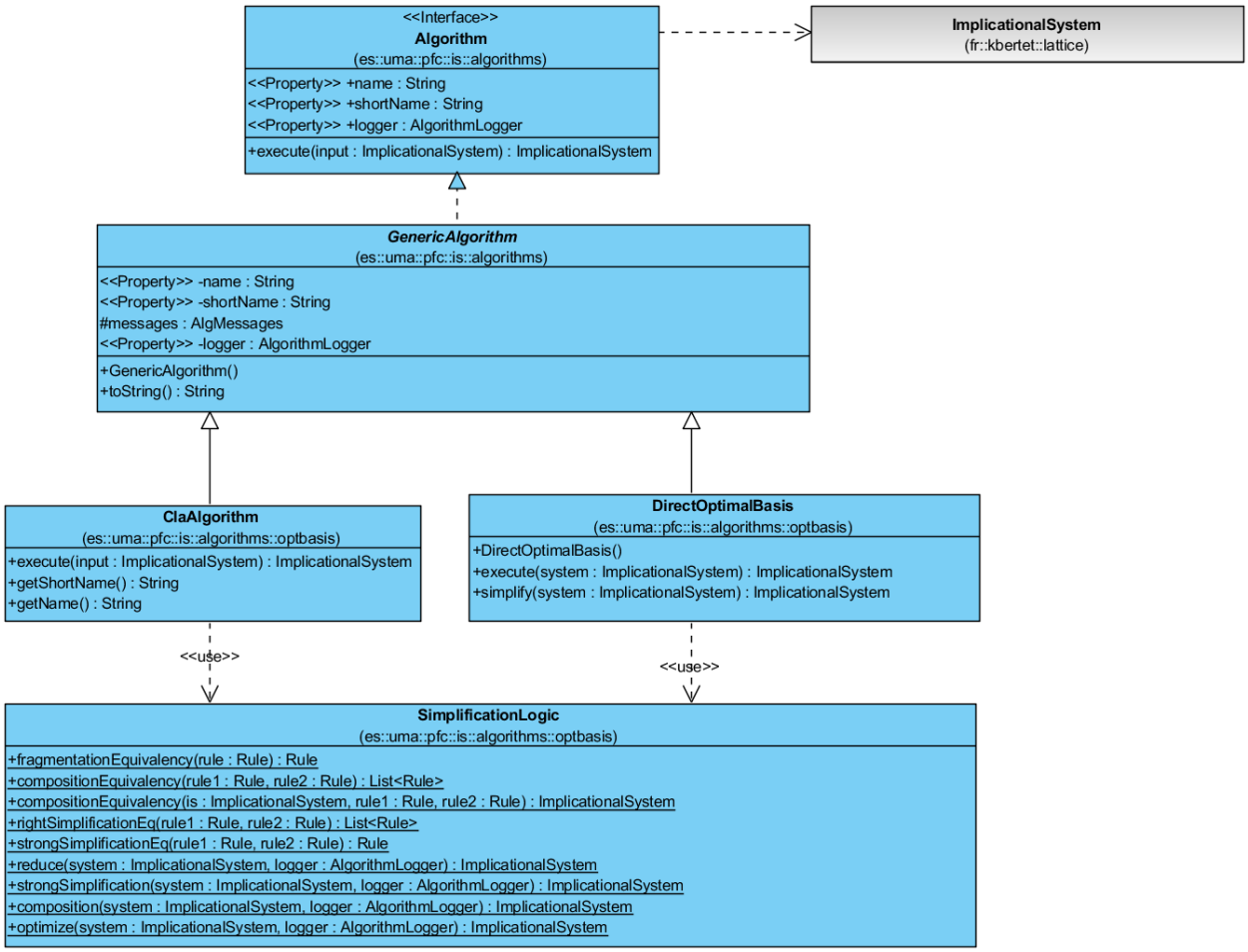
API de Algoritmos



Summary

Name	Description
Algorithm	Algorithm of implicational system basis computation.
GenericAlgorithm	Generic algorithm which receive as input a file input and returns an implicational system.
DirectOptimalBasis	AMIS Algorithm implementation.
ClaAlgorithm	CLA Algorithm implementation.
ImplicationalSystem	This class gives a representation for an implicational system (ImplicationalSystem), a set of rules. This class belong to java-lattices library.
SimplificationLogic	Methods which implements the simplification logic rules.

Description

API for implementation of algorithms which can be executed by IS Bench.

Details

Algorithm

Name	Value
Description	Algorithm of implicational system basis computation. This interface must be implemented by the classes which will be executed by IS Bench.
Visibility	public

Attributes

public name : String			
Description	Algorithm name. It will be used as the algorithm string representation.		
Stereotypes	Property		
Type	String		
Getter	`true`	Setter	`true`
Multiplicity	1		

public shortName : String			
Description	Algorithm short name. The short name will be used as the default name of the output files.		
Stereotypes	Property		
Type	String		
Getter	`true`	Setter	`true`
Multiplicity	1		

public logger : es.uma.pfc.is.logging.AlgorithmLogger			
Description	Logger for trace generation.		
Stereotypes	Property		
Type	es.uma.pfc.is.logging.AlgorithmLogger		
Getter	`true`	Setter	false
Multiplicity	1		

Operations

public execute (input : fr.kbertet.lattice.ImplicationalSystem) : fr.kbertet.lattice.ImplicationalSystem .ImplicationalSystem		
Parameters	**input**	
	Description	Input implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
Description	Executes the algorithm with the implicational system parameter as input.	
Return Type Description	Implicational System returned by the executed algorithm.	

GenericAlgorithm

Name	Value
Description	Generic algorithm which receive as input a file input and returns an implicational system.
Visibility	public

Attributes

private name : String			
Description	Name.		
Stereotypes	Property		
Type	String		
Getter	true	Setter	true
Multiplicity	1		

private shortName : String			
Description	Short name.		
Stereotypes	Property		
Type	String		
Getter	true	Setter	true
Multiplicity	1		

protected messages : AlgMessages			
Description	Translates messages to current language.		
Type	AlgMessages		
Getter	false	Setter	false
Multiplicity	1		

private logger : es.uma.pfc.is.logging.AlgorithmLogger			
Description	Logger.		
Stereotypes	Property		
Type	es.uma.pfc.is.logging.AlgorithmLogger		
Getter	true	Setter	false
Multiplicity	1		

Operations

public GenericAlgorithm ()	
Description	Constructor.

protected setLogger (logger : es.uma.pfc.is.logging.AlgorithmLogger) : void		
Parameters	**logger**	
	Description	Logger.
	Multiplicity	1
	Type	es.uma.pfc.is.logging.AlgorithmLogger
	Direction	inout
Description	For testing usage.	

protected removeRule (system : fr.kbertet.lattice.ImplicationalSystem, rule : fr.kbertet.lattice.Rule) : void		
Parameters	**system**	
	Description	Implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	rule	
	Description	Rule.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	Removes rules for implicational system, and print trace in the log.	

protected addRule (system : fr.kbertet.lattice.ImplicationalSystem, rule : fr.kbertet.lattice.Rule) : void		
Parameters	**system**	
	Description	Implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	rule	
	Description	Rule.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	Adds rules for implicational system, and print trace in the log.	
Return Type Description	Implicational system with a rule added.	

protected addRuleAndElements (system : fr.kbertet.lattice.ImplicationalSystem, rule : fr.kbertet.lattice.Rule) : ImplicationalSystem		
Parameters	**system**	
	Description	Implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	rule	
	Description	Rule.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	Adds rules for implicational system, and print trace in the log.	
Return Type Description	Implicational system with a rule and its elements added.	

protected addRuleAndElements (system : fr.kbertet.lattice.ImplicationalSystem, rule : fr.kbertet.lattice.Rule, trace : boolean) : fr.kbertet.lattice.ImplicationalSystem		
Parameters	**system**	
	Description	Implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	rule	
	Description	Rule.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
	trace	
	Description	If print trace in the log.
	Multiplicity	1
	Type	boolean
	Direction	inout
Description	Adds rules for implicational system and its elements, and print trace in the log if the trace parameter is true.	

protected history (message : String, args : Object) : void		
Parameters	**message**	
	Description	Message.
	Multiplicity	1
	Type	String
	Direction	inout
	args	
	Description	Message arguments.
	Multiplicity	0..*
	Type Modifier	...
	Type	Object
	Direction	inout
Description	Prints a message with the arguments, to the log.	

protected replaceRule (system : fr.kbertet.lattice.ImplicationalSystem, rule1 : fr.kbertet.lattice.Rule, rule2 : fr.kbertet.lattice.Rule) : void		
Parameters	**system**	
	Description	Implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	rule1	
	Description	Rule to replace.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
	rule2	
	Description	New rule.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	Replace a rule by other for implicational system, and print trace in the history.	

public toString () : String	
Description	Algorithm string representation. By default, is the name property value.
Return Type Description	Name.

DirectOptimalBasis

Name	Value
Description	Direct Optimal Basis algorithm implementation.
Visibility	public

Operations

public DirectOptimalBasis ()	
Description	Constructor.

public execute (system : ImplicationalSystem) : ImplicationalSystem		
Parameters	system	
	Description	Input system.
	Multiplicity	1
	Type	ImplicationalSystem
	Direction	inout
Description	Executes the Direct Optimal Basis algorithm.	

public simplify (system : ImplicationalSystem) : ImplicationalSystem		
Parameters	system	
	Description	Reduced system.
	Multiplicity	1
	Type	ImplicationalSystem
	Direction	inout
Description	Generation of IS simplificated by simplification(left+right+composition) of reduced IS	
Return Type Description	Simplified system.	
Query	false	

protected printInit (inputSystem : fr.kbertet.lattice.ImplicationalSystem) : void		
Parameters	inputSystem	
	Description	Implicactional System.
	Multiplicity	Unspecified
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
Description	Prints de initial arguments.	

protected printResult (resultSystem : fr.kbertet.lattice.ImplicationalSystem) : void		
Parameters	**resultSystem**	
	Description	Implicational System.
	Multiplicity	Unspecified
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
Description	Prints the results.	

ClaAlgorithm

Name	Value
Description	CLA Algorithm implementation.
Visibility	public

Operations

public execute (input : ImplicationalSystem) : ImplicationalSystem		
Parameters	**input**	
	Description	Input implicational system.
	Multiplicity	1
	Type	ImplicationalSystem
	Direction	inout
Description	Executes the CLA algorithm for computation of the input implicational system direct basis. Uses the SimplificationLogic class methods.	

public getShortName () : String	
Description	Short name.

public getName () : String	
Description	Name.

SimplificationLogic

Name	Value
Description	Methods which implements the simplification logic rules.
Visibility	public

Operations

public fragmentationEquivalency (implication : fr.kbertet.lattice.Rule) : fr.kbertet.lattice.Rule		
Parameters	**implication**	
	Description	Implication.
	Multiplicity	1
	Type	Rule
	Direction	inout
Description	Implements the Fragmentation Equivalency rule: [FrEq]: {A -> B} = {A -> B-A}.	
Return Type Description	Equivalent simplified implication.	

public compositionEquivalency (rule1 : fr.kbertet.lattice.Rule, rule2 : fr.kbertet.lattice.Rule) :java.util.List< fr.kbertet.lattice.Rule>		
Parameters	**rule1**	
	Description	Implication.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
	rule2	
	Description	Implication.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	Given two implications: {A -> B, A -> C} => {A -> BC}	
Return Type Description	If the composition rule can be applied, returns a list with once implication from the rule application. If the composition rule can’t be applied, returns a list which	

public compositionEquivalency (rule1 : fr.kbertet.lattice.Rule, rule2 : fr.kbertet.lattice.Rule) :java.util.List< fr.kbertet.lattice.Rule>	
	contains the rules passed as parameters.
Exceptions	java.lang.NullPointerException Si alguna de las implicaciones es nula.

public compositionEquivalency (is : fr.kbertet.lattice.ImplicationalSystem, rule1 : fr.kbertet.lattice.Rule, rule2 : fr.kbertet.lattice.Rule) : fr.kbertet.lattice.ImplicationalSystem		
Parameters	**is**	
	Description	Implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	rule1	
	Description	Implication.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
	rule2	
	Description	Implication.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	rule1 and rule2 are implications from is implicational system parameter. If the composition rule can be applied to rule1 and rule2, removes these rules from the implicational system and adds the new.	

public rightSimplificationEq (rule1 : fr.kbertet.lattice.Rule, rule2 : fr.kbertet.lattice. Rule) : java.util.List< fr.kbertet.lattice.>		
Parameters	**rule1**	
	Description	Implication.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
	rule2	
	Description	Implication.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	Implements the simplification rule for two implications. [SiEq]: if (A intersection B is empty) and (A subset of C) then {A -> B, C -> D} == {A -> B, C-B -> D-B}	
Return Type Description	Simplified implications.	
Exceptions	java.lang.NullPointerException if any rule is null.	

public strongSimplificationEq (rule1 : fr.kbertet.lattice.Rule, rule2 : fr.kbertet.lattice.Rule) : fr.kbertet.lattice.Rule		
Parameters	**rule1**	
	Description	Implication.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
	rule2	
	Description	Implicación.
	Multiplicity	1
	Type	fr.kbertet.lattice.Rule
	Direction	inout
Description	If (B intersection C) not is empty and (D \ A union B)) neither,	

public strongSimplificationEq (rule1 : fr.kbertet.lattice.Rule, rule2 : fr.kbertet.lattice.Rule) : fr.kbertet.lattice.Rule	
	returns the new implication AC - B -> D - (AB).
Return Type Description	Returns a new implication if Strong Simplification can be applied, null otherwise.

public reduce (system : ImplicationalSystem, logger : AlgorithmLogger) : fr.kbertet.lattice.ImplicationalSystem		
Parameters	**system**	
	Description	Implicational system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	logger	
	Description	Logger.
	Multiplicity	1
	Type	AlgorithmLogger
	Direction	inout
Description	Gets a reduced system. An implicational system is reduced, if A -> B => (B not empty) AND (A intersection B is empty) for all A, B in S.	
Return Type Description	Reduced implicational system.	

public strongSimplification (system : fr.kbertet.lattice.ImplicationalSystem, logger : AlgorithmLogger) : fr.kbertet.lattice.ImplicationalSystem		
Parameters	**system**	
	Description	Simplified system.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	logger	
	Description	Logger.
	Multiplicity	1
	Type	AlgorithmLogger
	Direction	inout
Description	Generation of IS by completion of simplified IS --> Strong Simplification.	
Return Type Description	Strong simplified system.	
Exceptions	java.lang.NullPointerException if system is null.	

public composition (system : fr.kbertet.lattice.ImplicationalSystem, logger : AlgorithmLogger) : fr.kbertet.lattice. ImplicationalSystem		
Parameters	**system**	
	Description	Implicational System.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	logger	
	Descriptoin	Loger
	Multiplicity	1
	Type	AlgorithmLogger
	Direction	inout
Description	Composition of implications of a system.	
Return Type Description	Implicational System with composition applied.	

public optimize (system : fr.kbertet.lattice.ImplicationalSystem, logger : AlgorithmLogger) : fr.kbertet.lattice.ImplicationalSystem		
Parameters	**system**	
	Description	Simplified Implicational System.
	Multiplicity	1
	Type	fr.kbertet.lattice.ImplicationalSystem
	Direction	inout
	logger	
	Description	Logger.
	Multiplicity	1
	Type	AlgorithmLogger
	Direction	inout
Description	Generation of optimized IS.	
Return Type Description	Optimized system.	