

ISBJava API Documentation

CompBio Group, Institute for Systems Biology

Version 2.4.4, Released 2006/09/28

Please e-mail comments or corrections to: `isbjava at systemsbiology.org`

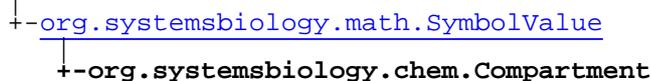
Package
org.systemsbiology.chem

/p>

org.systemsbiology.chem

Class Compartment

java.lang.Object

public final class **Compartment**extends [SymbolValue](#)

Represents a named, well-mixed reaction compartment, which has a numeric volume. In Model objects constructed using the {@link ModelBuilderCommandLanguage}, typically only a default Compartment object (with unit volume) is used, with the actual volume being absorbed into the values for the reaction parameters. However, Model objects constructed from SBML using the {@link org.systemsbiology.chem.sbml.ModelBuilderMarkupLanguage} are special, in that the symbol evaluator implicitly makes use of the Compartment volume when evaluating a Species symbol that appears in a `kineticLaw` formula.

Fields

DEFAULT_VOLUME

public static final double **DEFAULT_VOLUME**

Constructors

Compartment

```
public Compartment(java.lang.String pName,
                   double pVolume)
```

Creates a compartment. The compartment name may not contain the NAME_DELIMITER string, which is a single colon character.

Compartment

```
public Compartment(SymbolValue pSymbolValue)
```

Compartment

```
public Compartment(java.lang.String pName)
```

Methods

setVolume

```
public void setVolume(double pVolume)
```

(continued on next page)

(continued from last page)

setVolume

```
public void setVolume(Expression pVolume)
```

getName

```
public java.lang.String getName()
```

equals

```
public boolean equals(Compartment pCompartment)
```

toString

```
public java.lang.String toString()
```

clone

```
public java.lang.Object clone()
```

org.systemsbiology.chem

Class Constants

java.lang.Object

└-org.systemsbiology.chem.Constants

public final class **Constants**

extends java.lang.Object

Definitions of constants that are useful in simulations of coupled chemical reactions.

Fields

AVOGADRO_CONSTANT

public static final double **AVOGADRO_CONSTANT**

The number of molecules per mole. Value taken from the 1998 CODATA recommended values. Uncertainty is plus or minus 0.00000047e+23.

Constructors

Constants

public **Constants**()

org.systemsbiology.chem

Class DelayedReactionSolver

java.lang.Object

└--org.systemsbiology.chem.DelayedReactionSolver

```
public final class DelayedReactionSolver
extends java.lang.Object
```

Used to simulate a chemical reaction containing a specified delay time. The reactant is immediately converted to a (hidden) "intermediate species". The reaction converting the intermediate species to the product species occurs after the specified delay. This class is used by subclasses of the { @link Simulator } class. The application developer will rarely need to work directly with an instance of this class.

Fields

MIN_NUM_HISTORY_BINS

```
public static final int MIN_NUM_HISTORY_BINS
```

DEFAULT_NUM_HISTORY_BINS

```
public static final int DEFAULT_NUM_HISTORY_BINS
```

Constructors

DelayedReactionSolver

```
public DelayedReactionSolver(Species pReactant,
                             Species pIntermedSpecies,
                             double pDelay,
                             double pRate,
                             boolean pIsMultistep,
                             int pReactionIndex,
                             boolean pIsStochasticSimulator)
```

Methods

toString

```
public java.lang.String toString()
```

getRate

```
public double getRate()
```

(continued from last page)

getDelay

```
public double getDelay()
```

getIntermedSpecies

```
public Species getIntermedSpecies()
```

getNumHistoryBins

```
public int getNumHistoryBins()
```

setNumHistoryBins

```
public void setNumHistoryBins(int pNumHistoryBins)
```

update

```
public void update(SymbolEvaluator pSymbolEvaluator,  
                  double pTime)  
    throws DataNotFoundException
```

computeRate

```
public double computeRate(SymbolEvaluator pSymbolEvaluator)  
    throws DataNotFoundException
```

org.systemsbiology.chem

Interface IModelBuilder

public interface **IModelBuilder**

Represents an object that can build a {@link Model} from an InputStream. Typically, the input stream is a file or a string containing a model description. The {@link org.systemsbiology.util.IncludeHandler} is typically only used for file-based model descriptions. The application developer typically obtains an instance of a model builder corresponding to the type of model description file (SBML, CMDL, etc.), and passes the model description to the {@link #buildModel(InputStream, IncludeHandler)} method as an InputStream. The {@link Model} instance that was built, is returned from this method. All subclasses of this interface should implement {@link org.systemsbiology.util.IAliasableClass}, and have the public static string field `CLASS_ALIAS`.

Methods

buildModel

```
public Model buildModel (java.io.InputStream pInputStream,
                        IncludeHandler pIncludeHandler)
    throws InvalidInputException,
           java.io.IOException
```

getFileRegex

```
public java.lang.String getFileRegex()
```

readModel

```
public java.lang.String readModel (java.io.InputStream pInputStream)
    throws InvalidInputException,
           java.io.IOException
```

writeModel

```
public void writeModel (java.lang.String pModelText,
                       java.io.OutputStream pOutputStream)
    throws InvalidInputException
```

Write the model to the specified OutputStream, using the correct character encoding.

org.systemsbiology.chem

Interface IModelExporter

public interface **IModelExporter**

Defines a class that can write a markup language description of a {@link Model} containing a set of chemical {@link Reaction} objects and the the initial populations of the chemical {@link Species}. All subclasses of this interface should implement {@link org.systemsbiology.util.IAliasableClass}, and have the public static string field CLASS_ALIAS.

See Also:

[Model](#), [Species](#)

Methods

export

```
public void export(Model pModel,  
                  java.io.PrintWriter pOutputWriter)  
    throws java.lang.IllegalArgumentException,  
           DataNotFoundException,  
           java.lang.IllegalStateException,  
           java.lang.UnsupportedOperationException,  
           ModelExporterException
```

getFileRegex

```
public java.lang.String getFileRegex()
```

org.systemsbiology.chem

Interface IModelViewer

public interface **IModelViewer**

Represents an object that can display a {@link Model} in graphical or textual format, on the screen. All subclasses of this interface should implement {@link org.systemsbiology.util.IAliasableClass}, and have the public static string field CLASS_ALIAS.

Methods

viewModel

```
public void viewModel(Model pModel,  
                      java.lang.String pAppName)  
    throws ModelViewerException
```

org.systemsbiology.chem

Interface ISimulator

```
public interface ISimulator
```

Represents an object that can solve the dynamics of a {@link Model} over time. All subclasses of this interface should implement {@link org.systemsbiology.util.IAliasableClass}, and have the public static string field `CLASS_ALIAS`.

Fields

MIN_NUM_RESULTS_TIME_POINTS

```
public static final int MIN_NUM_RESULTS_TIME_POINTS
```

Methods

getAlias

```
public java.lang.String getAlias()
```

isInitialized

```
public boolean isInitialized()
```

initialize

```
public void initialize(Model pModel)
    throws DataNotFoundException,
           InvalidInputException
```

setProgressReporter

```
public void setProgressReporter(SimulationProgressReporter
    pSimulationProgressReporter)
```

setController

```
public void setController(SimulationController pSimulationController)
```

checkSimulationParameters

```
public void checkSimulationParameters(double pStartTime,
    double pEndTime,
    SimulatorParameters pSimulatorParameters,
    int pNumResultsTimePoints)
```

getDefaultSimulatorParameters

```
public SimulatorParameters getDefaultSimulatorParameters()
```

setStatusUpdateIntervalSeconds

```
public void setStatusUpdateIntervalSeconds(double pStatusUpdateIntervalSeconds)
    throws java.lang.IllegalArgumentException
```

Set the update interval, in seconds. The updates are provided through the {@link SimulationProgressReporter} class. This also sets the interval for checking the {@link SimulationController} for pause or cancellation. The value specified must be greater than zero. If this method is not called, a default value of 1.0 seconds is used.

allowsInterrupt

```
public boolean allowsInterrupt()
```

canComputeFluctuations

```
public boolean canComputeFluctuations()
```

simulate

```
public SimulationResults simulate(double pStartTime,
    double pEndTime,
    SimulatorParameters pSimulatorParameters,
    int pNumResultsTimePoints,
    java.lang.String pResultsSymbolNames)
    throws DataNotFoundException,
    java.lang.IllegalStateException,
    java.lang.IllegalArgumentException,
    AccuracyException,
    SimulationFailedException
```

Conduct a simulation of the dynamics of the {@link Model} passed to the {@link #initialize(Model)} method. The end time must be greater than the start time. The integer pNumResultsTimePoints must be greater than zero. The size of the results arrays must be equal to one plus pNumResultsTimePoints (the extra element in the arrays is to hold the initial data). The parameter pNumResultsTimePoints must be greater than or equal to 2.

org.systemsbiology.chem

Class Model

java.lang.Object

└-org.systemsbiology.chem.Model

public final class **Model**

extends java.lang.Object

A named collection of { @link Reaction } and { @link Parameter } objects, which represents a system of interacting chemical { @link Species }. The chemical Species are contained in the Reaction objects.

Fields

INTERNAL_SYMBOL_PREFIX

public static final java.lang.String **INTERNAL_SYMBOL_PREFIX**

NAMESPACE_IDENTIFIER

public static final java.lang.String **NAMESPACE_IDENTIFIER**

Constructors

Model

public **Model**()

Model

public **Model**(java.lang.String pName)

Methods

getReservedSymbolMapper

public [ReservedSymbolMapper](#) **getReservedSymbolMapper**()

setReservedSymbolMapper

public void **setReservedSymbolMapper**(ReservedSymbolMapper pReservedSymbolMapper)

Associate a { @link ReservedSymbolMapper } with this Model. This allows { @link org.systemsbiology.math.Expression expressions } that reference reserved symbols from the ReservedSymbolMapper. For example, in order to have a model in which expressions can reference the symbol "time", you need to create an instance of { @link ReservedSymbolMapperChemCommandLanguage } and pass it to this method.

(continued from last page)

Parameters:

pReservedSymbolMapper

setSymbolEvaluationPostProcessor

```
public void setSymbolEvaluationPostProcessor(SymbolEvaluationPostProcessor  
pSymbolEvaluationPostProcessor)
```

getSymbolEvaluationPostProcessor

```
public SymbolEvaluationPostProcessor getSymbolEvaluationPostProcessor()
```

setName

```
public void setName(java.lang.String pName)
```

Sets the model name to be the string contained in pName This overrides the model name passed to the constructor.

getName

```
public java.lang.String getName()
```

addParameter

```
public void addParameter(Parameter pParameter)
```

addSpecies

```
public void addSpecies(Species pSpecies)
```

getDynamicSymbols

```
public java.util.Collection getDynamicSymbols()
```

getReactions

```
public java.util.Collection getReactions()
```

getSymbols

```
public java.util.Collection getSymbols()
```

addReaction

```
public void addReaction(Reaction pReaction)  
    throws java.lang.IllegalStateException
```

It is illegal to add the reaction with a given name, twice

getSpeciesByName

```
public Species getSpeciesByName(java.lang.String pSpeciesName)  
    throws DataNotFoundException
```

containsDelayedOrMultistepReaction

```
public boolean containsDelayedOrMultistepReaction()
```

getOrderedSpeciesNamesArray

```
public java.lang.String[] getOrderedSpeciesNamesArray()
```

getOrderedResultsSymbolNamesArray

```
public java.lang.String[] getOrderedResultsSymbolNamesArray()  
    throws  
java.lang.IllegalStateException
```

toString

```
public java.lang.String toString()
```

org.systemsbiology.chem

Class ModelBuilderCommandLanguage

java.lang.Object

└--org.systemsbiology.chem.ModelBuilderCommandLanguage

All Implemented interfaces:

[IAliasableClass](#), [IModelBuilder](#)

```
public class ModelBuilderCommandLanguage
extends java.lang.Object
implements IModelBuilder, IAliasableClass
```

Builds a { @link Model } object from a CMDL model description.

Fields

CLASS_ALIAS

public static final java.lang.String CLASS_ALIAS

Constructors

ModelBuilderCommandLanguage

public ModelBuilderCommandLanguage()

Methods

buildModel

```
public Model buildModel(java.io.InputStream pInputStream,
                        IncludeHandler pIncludeHandler)
    throws InvalidInputException,
           java.io.IOException
```

writeModel

```
public void writeModel(java.lang.String pModelText,
                       java.io.OutputStream pOutputStream)
```

getBufferedReader

public java.io.BufferedReader getBufferedReader(java.io.InputStream pInputStream)

(continued from last page)

readModel

```
public java.lang.String readModel(java.io.InputStream pInputStream)
    throws java.io.IOException
```

isValidSymbol

```
public static boolean isValidSymbol(java.lang.String pSymbolName)
```

getFileRegex

```
public java.lang.String getFileRegex()
```

org.systemsbiology.chem

Class ModelExporterCommandLanguage

java.lang.Object

└-org.systemsbiology.chem.ModelExporterCommandLanguage

All Implemented interfaces:

[IAliasableClass](#), [IModelExporter](#)

```
public class ModelExporterCommandLanguage
extends java.lang.Object
implements IModelExporter, IAliasableClass
```

Saves a {@link org.systemsbiology.chem.Model} in the CMDL format.

Fields

CLASS_ALIAS

public static final java.lang.String CLASS_ALIAS

Constructors

ModelExporterCommandLanguage

public ModelExporterCommandLanguage()

Methods

export

```
public void export(Model pModel,
                  java.io.PrintWriter pOutputWriter)
throws java.lang.IllegalArgumentException,
      DataNotFoundException,
      java.lang.IllegalStateException,
      java.lang.UnsupportedOperationException,
      ModelExporterException
```

Given a {@link org.systemsbiology.chem.Model} object defining a system of chemical reactions and the initial species populations, writes out the model in Chemical Model Definition Language (CMDL)

getFileRegex

public java.lang.String getFileRegex()

org.systemsbiology.chem

Class **ModelExporterException**

```
java.lang.Object
  |
  +- java.lang.Throwable
      |
      +- java.lang.Exception
          |
          +- org.systemsbiology.chem.ModelExporterException
```

```
public class ModelExporterException
  extends java.lang.Exception
```

Constructors

ModelExporterException

```
public ModelExporterException(java.lang.String pMessage)
```

ModelExporterException

```
public ModelExporterException(java.lang.String pMessage,
                               java.lang.Throwable pCause)
```

org.systemsbiology.chem

Class ModelExporterHumanReadable

java.lang.Object

└--org.systemsbiology.chem.ModelExporterHumanReadable

All Implemented interfaces:

[IAliasableClass](#), [IModelExporter](#)

```
public class ModelExporterHumanReadable
extends java.lang.Object
implements IModelExporter, IAliasableClass
```

Writes the model output in human-readable format

Fields

CLASS_ALIAS

public static final java.lang.String CLASS_ALIAS

Constructors

ModelExporterHumanReadable

public ModelExporterHumanReadable()

Methods

export

```
public void export(Model pModel,
                  java.io.PrintWriter pOutputWriter)
throws java.lang.IllegalArgumentException,
      DataNotFoundException,
      java.lang.IllegalStateException,
      java.lang.UnsupportedOperationException,
      ModelExporterException
```

Given a {@link org.systemsbiology.chem.Model} object defining a system of chemical reactions and the initial species populations, writes out the model in the column format specified by David Orrell

getFileRegex

public java.lang.String getFileRegex()

org.systemsbiology.chem

Class ModelExporterOrrellColumnFormat

java.lang.Object

└--org.systemsbiology.chem.ModelExporterOrrellColumnFormat

All Implemented interfaces:

[IAliasableClass](#), [IModelExporter](#)

public class **ModelExporterOrrellColumnFormat**

extends java.lang.Object

implements [IModelExporter](#), [IAliasableClass](#)

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

Constructors

ModelExporterOrrellColumnFormat

public **ModelExporterOrrellColumnFormat**()

Methods

export

```
public void export(Model pModel,
                   java.io.PrintWriter pOutputWriter)
    throws java.lang.IllegalArgumentException,
           DataNotFoundException,
           java.lang.IllegalStateException,
           java.lang.UnsupportedOperationException,
           ModelExporterException
```

Given a {@link org.systemsbiology.chem.Model} object defining a system of chemical reactions and the initial species populations, writes out the model in the column format specified by David Orrell

getFileRegex

```
public java.lang.String getFileRegex()
```

org.systemsbiology.chem

Class **ModelViewerException**

```
java.lang.Object
  |
  +- java.lang.Throwable
    |
    +- java.lang.Exception
      |
      +- org.systemsbiology.chem.ModelViewerException
```

```
public class ModelViewerException
  extends java.lang.Exception
```

Constructors

ModelViewerException

```
public ModelViewerException(java.lang.String pMessage)
```

ModelViewerException

```
public ModelViewerException(java.lang.String pMessage,
                             java.lang.Throwable pCause)
```

org.systemsbiology.chem

Class Parameter

java.lang.Object

```
├--org.systemsbiology.math.SymbolValue
└--org.systemsbiology.chem.Parameter
```

```
public final class Parameter
extends SymbolValue
```

Constructors

Parameter

```
public Parameter(java.lang.String pName,
                 Expression pValue)
```

Parameter

```
public Parameter(java.lang.String pName)
```

Parameter

```
public Parameter(java.lang.String pName,
                 double pValue)
```

Parameter

```
public Parameter(SymbolValue pSymbolValue)
```

Methods

getSymbolName

```
public java.lang.String getSymbolName()
```

getName

```
public java.lang.String getName()
```

setValue

```
public void setValue(Expression pValue)
```

(continued from last page)

setValue

```
public void setValue(double pValue)
```

clone

```
public java.lang.Object clone()
```

toString

```
public java.lang.String toString()
```

org.systemsbiology.chem

Class Reaction

```
java.lang.Object
  |
  +--org.systemsbiology.math.SymbolValue
      |
      +--org.systemsbiology.chem.Reaction
```

```
public final class Reaction
extends SymbolValue
```

Represents a chemical process that can take place, transforming zero or more distinct reactant {[@link Species](#)} into zero or more distinct product {[@link Species](#)}. Each species that participates in a Reaction has an integer stoichiometry, specifying the number of molecules of that species that are consumed or produced in the reaction. Reactions are typically constructed, and then added to a {[@link Model](#)} object. Zero reactants **and** zero product species is a degenerate case that is not allowed. A Species that participates in a Reaction is described internally using a {[@link ReactionParticipant](#)} object that specifies the Species and stoichiometry. Reactions are typically defined with a floating-point rate, which defines the reaction parameter. The average rate at which the reaction is occurring is the product of the reaction parameter, and the number of distinct combinations of reactant molecules. Alternatively, a reaction may have its rate defined in terms of an {[@link org.systemsbiology.math.Expression](#)}, which is an algebraic expression involving various {[@link org.systemsbiology.math.Symbol](#)} names. Such symbol names may represent {[@link Parameter](#)}, {[@link Compartment](#)}, or {[@link Species](#)} objects. Note that in this case (using an Expression to define the reaction rate), the expression is simply evaluated to obtain the rate of the reaction; there is no post-multiplication by the number of reactant combinations.

Constructors

Reaction

```
public Reaction(java.lang.String pName)
```

Methods

clone

```
public java.lang.Object clone()
```

getParameters

```
public java.util.Collection getParameters()
```

addParameter

```
public void addParameter(Parameter pParameter)
```

hasLocalSymbols

```
public boolean hasLocalSymbols()
```

setNumSteps

```
public void setNumSteps(int pNumSteps)
```

getDelay

```
public double getDelay()
```

setDelay

```
public void setDelay(double pDelay)
```

getNumSteps

```
public int getNumSteps()
```

getNumParticipants

```
public int getNumParticipants(ReactionParticipant.Type pParticipantType)
```

constructSpeciesArrays

```
public void constructSpeciesArrays(Species pSpeciesArray,  
                                   int pStoichiometryArray,  
                                   boolean pDynamicArray,  
                                   ReactionParticipant.Type pParticipantType)
```

getNumReactants

```
public int getNumReactants()
```

getNumProducts

```
public int getNumProducts()
```

getNumLocalSymbols

```
public int getNumLocalSymbols()
```

equals

```
public boolean equals(Reaction pReaction)
```

getName

```
public java.lang.String getName()
```

setRate

```
public void setRate(Value pRate)
```

setRate

```
public void setRate(double pRate)
```

setRate

```
public void setRate(Expression pRate)
```

addReactionParticipantToMap

```
public void addReactionParticipantToMap(ReactionParticipant pReactionParticipant,  
                                         java.util.HashMap pMap)  
    throws java.lang.IllegalArgumentException
```

addReactant

```
public void addReactant(Species pSpecies,  
                        int pStoichiometry,  
                        boolean pDynamic)  
    throws java.lang.IllegalStateException
```

addReactant

```
public void addReactant(Species pSpecies,  
                        int pStoichiometry)  
    throws java.lang.IllegalStateException
```

addProduct

```
public void addProduct(Species pSpecies,  
                       int pStoichiometry)  
    throws java.lang.IllegalStateException
```

addProduct

```
public void addProduct(Species pSpecies,  
                       int pStoichiometry,  
                       boolean pDynamic)  
    throws java.lang.IllegalStateException
```

(continued from last page)

addSpecies

```
public void addSpecies(Species pSpecies,  
                        int pStoichiometry,  
                        boolean pDynamic,  
                        ReactionParticipant.Type pParticipantType)  
    throws java.lang.IllegalArgumentException
```

toString

```
public java.lang.String toString()
```

org.systemsbiology.chem

Class ReactionParticipant

java.lang.Object

└─org.systemsbiology.chem.ReactionParticipant

All Implemented interfaces:

java.lang.Comparable

public final class **ReactionParticipant**

extends java.lang.Object

implements java.lang.Comparable

Constructors

ReactionParticipant

```
public ReactionParticipant(Species pSpecies,  
                           int pStoichiometry,  
                           boolean pDynamic)
```

Methods

getStoichiometry

```
public int getStoichiometry()
```

getDynamic

```
public boolean getDynamic()
```

getSpecies

```
public Species getSpecies()
```

equals

```
public boolean equals(ReactionParticipant pReactionParticipant)
```

compareTo

```
public int compareTo(java.lang.Object pReactionParticipant)
```

toString

```
public java.lang.String toString()
```

org.systemsbiochemistry

Class ReactionParticipant.Type

java.lang.Object

└-org.systemsbiochemistry.ReactionParticipant.Type

public static final class **ReactionParticipant.Type**
extends java.lang.Object

Fields

REACTANT

public static final org.systemsbiochemistry.ReactionParticipant.Type **REACTANT**

PRODUCT

public static final org.systemsbiochemistry.ReactionParticipant.Type **PRODUCT**

Methods

toString

public java.lang.String **toString()**

org.systemsbiology.chem

Class **ReservedSymbolMapperChemCommandLanguage**

java.lang.Object

├─[org.systemsbiology.math.ReservedSymbolMapper](#)

└─org.systemsbiology.chem.ReservedSymbolMapperChemCommandLanguage

```
public final class ReservedSymbolMapperChemCommandLanguage
extends ReservedSymbolMapper
```

Fields

SYMBOL_TIME

```
public static final java.lang.String SYMBOL_TIME
```

SYMBOL_AVOGADRO

```
public static final java.lang.String SYMBOL_AVOGADRO
```

Constructors

ReservedSymbolMapperChemCommandLanguage

```
public ReservedSymbolMapperChemCommandLanguage()
```

Methods

getReservedSymbolNames

```
public java.util.Collection getReservedSymbolNames()
```

isReservedSymbol

```
public static boolean isReservedSymbol(java.lang.String pSymbolName)
```

isReservedSymbol

```
public boolean isReservedSymbol(Symbol pSymbol)
```

(continued from last page)

getReservedSymbolValue

```
public double getReservedSymbolValue(Symbol pSymbol,  
                                     SymbolEvaluator pSymbolEvaluator)  
    throws DataNotFoundException
```

org.systemsbiology.chem

Class SimulationController

java.lang.Object

└--org.systemsbiology.chem.SimulationController

```
public final class SimulationController
extends java.lang.Object
```

A data structure that allows starting and stopping a simulation while it is in progress. This class is intended to be thread-safe.

See Also:

[ISimulator](#)

Constructors

SimulationController

```
public SimulationController()
```

Methods

setCancelled

```
public void setCancelled(boolean pCancelled)
```

getCancelled

```
public boolean getCancelled()
```

setStopped

```
public void setStopped(boolean pStopped)
```

getStopped

```
public boolean getStopped()
```

handlePauseOrCancel

```
public boolean handlePauseOrCancel()
```

org.systemsbiology.chem

Class **SimulationFailedException**

```
java.lang.Object
  |
  +- java.lang.Throwable
      |
      +- java.lang.Exception
          |
          +- org.systemsbiology.chem.SimulationFailedException
```

```
public class SimulationFailedException
  extends java.lang.Exception
```

Constructors

SimulationFailedException

```
public SimulationFailedException(java.lang.String pMessage)
```

SimulationFailedException

```
public SimulationFailedException(java.lang.String pMessage,
                                 java.lang.Throwable pCause)
```

org.systemsbiology.chem

Class SimulationProgressReporter

java.lang.Object

└-org.systemsbiology.chem.SimulationProgressReporter

```
public final class SimulationProgressReporter
extends java.lang.Object
```

Constructors

SimulationProgressReporter

```
public SimulationProgressReporter()
```

Methods

setSimulationFinished

```
public void setSimulationFinished(boolean pSimulationFinished)
```

getSimulationFinished

```
public boolean getSimulationFinished()
```

waitForUpdate

```
public void waitForUpdate()
```

getIterationCounter

```
public long getIterationCounter()
```

updateProgressStatistics

```
public void updateProgressStatistics(boolean pSimulationFinished,
                                     double pFractionComplete,
                                     long pIterationCounter)
```

getTimeOfLastUpdateMillis

```
public long getTimeOfLastUpdateMillis()
```

getFractionComplete

```
public double getFractionComplete()
```

org.systemsbiology.chem

Class SimulationResults

java.lang.Object

└-org.systemsbiology.chem.SimulationResults

public final class **SimulationResults**

extends java.lang.Object

Data structure that contains the results of a simulation. This includes the time range for the simulation, the name of the simulator used, the name of the chemical species to be viewed, etc.

Constructors

SimulationResults

public **SimulationResults**()

Methods

setModelName

public void **setModelName**(java.lang.String pModelName)

getModelName

public java.lang.String **getModelName**()

getResultsFinalSymbolFluctuations

public double[] **getResultsFinalSymbolFluctuations**()

setResultsFinalSymbolFluctuations

public void **setResultsFinalSymbolFluctuations**(double pResultsFinalSymbolFluctuations)

getSimulatorAlias

public java.lang.String **getSimulatorAlias**()

getStartTime

public double **getStartTime**()

getEndTime

```
public double getEndTime()
```

getSimulatorParameters

```
public SimulatorParameters getSimulatorParameters()
```

getResultsSymbolNames

```
public java.lang.String[] getResultsSymbolNames()
```

Returns an array containing the names of the symbols for which the user requested to view the time-series data results for the simulation.

getResultsTimeValues

```
public double[] getResultsTimeValues()
```

Returns an array containing the time values of the time points at which the symbols (requested by the user) were evaluated.

getResultsSymbolValues

```
public java.lang.Object[] getResultsSymbolValues()
```

A two-dimensional array of doubles containing the actual values of the symbols requested by the user. The first index identifies the time point, and returns an array of doubles. That array of doubles is of the same length as the number of symbols requested by the user; it contains the values of the corresponding symbols at the time point identified by the first array index; schematically, access would look like this:

```
value = ((double [])resultsSymbolValues[timeIndex])[symbolIndex]
```

setSimulatorAlias

```
public void setSimulatorAlias(java.lang.String pSimulatorAlias)
```

setStartTime

```
public void setStartTime(double pStartTime)
```

setEndTime

```
public void setEndTime(double pEndTime)
```

setSimulatorParameters

```
public void setSimulatorParameters(SimulatorParameters pSimulatorParameters)
```

setResultsSymbolNames

```
public void setResultsSymbolNames(java.lang.String pResultsSymbolNames)
```

(continued from last page)

setResultsTimeValues

```
public void setResultsTimeValues(double pResultsTimeValues)
```

setResultsSymbolValues

```
public void setResultsSymbolValues(java.lang.Object pResultsSymbolValues)
```

getResultsDateTime

```
public java.util.Date getResultsDateTime()
```

createLabel

```
public java.lang.String createLabel()
```

org.systemsbiology.chem

Class Simulator

java.lang.Object

└-org.systemsbiology.chem.Simulator

Direct Known Subclasses:[SimulatorOdeToJavaBase](#), [SimulatorStochasticBase](#), [SimulatorDeterministicBase](#)public abstract class **Simulator**

extends java.lang.Object

Base class extended by all classes implementing {@link ISimulator}. This class contains a lot of methods and fields of common use to all simulator classes. From the standpoint of object-oriented correctness, this really should be done by composition rather than inheritance. For performance, accessor methods are avoided in favor of direct field access by subclasses.

Fields

DEFAULT_MIN_NUM_MILLISECONDS_FOR_UPDATE

protected static final long **DEFAULT_MIN_NUM_MILLISECONDS_FOR_UPDATE**

NULL_REACTION

protected static final int **NULL_REACTION**

mModelName

protected java.lang.String **mModelName**

mNonDynamicSymbolValues

protected org.systemsbiology.math.Value **mNonDynamicSymbolValues**

mNonDynamicSymbolExpressionValues

protected org.systemsbiology.math.Value **mNonDynamicSymbolExpressionValues**

mUseExpressionValueCaching

protected boolean **mUseExpressionValueCaching**

(continued from last page)

mSymbolEvaluator

protected org.systemsbiology.chem.SymbolEvaluatorChem **mSymbolEvaluator**

mSymbolMap

protected java.util.HashMap **mSymbolMap**

mSimulationController

protected org.systemsbiology.chem.SimulationController **mSimulationController**

mDelayedReactionSolvers

protected org.systemsbiology.chem.DelayedReactionSolver **mDelayedReactionSolvers**

mInitialized

protected boolean **mInitialized**

mMinNumMillisecondsForUpdate

protected long **mMinNumMillisecondsForUpdate**

mSimulationProgressReporter

protected org.systemsbiology.chem.SimulationProgressReporter
mSimulationProgressReporter

mIsStochasticSimulator

protected boolean **mIsStochasticSimulator**

mDynamicSymbols

protected org.systemsbiology.chem.Species **mDynamicSymbols**

mInitialDynamicSymbolValues

protected double **mInitialDynamicSymbolValues**

mDynamicSymbolNames

protected java.lang.String **mDynamicSymbolNames**

(continued from last page)

mDynamicSymbolValues

protected double **mDynamicSymbolValues**

mReactions

protected org.systemsbiology.chem.Reaction **mReactions**

mDynamicSymbolAdjustmentVectors

protected java.lang.Object **mDynamicSymbolAdjustmentVectors**

mReactionProbabilities

protected double **mReactionProbabilities**

mReactionsReactantsSpecies

protected java.lang.Object **mReactionsReactantsSpecies**

mReactionsReactantsStoichiometries

protected java.lang.Object **mReactionsReactantsStoichiometries**

mReactionsReactantsDynamic

protected java.lang.Object **mReactionsReactantsDynamic**

mReactionsProductsSpecies

protected java.lang.Object **mReactionsProductsSpecies**

mReactionsProductsStoichiometries

protected java.lang.Object **mReactionsProductsStoichiometries**

mReactionsProductsDynamic

protected java.lang.Object **mReactionsProductsDynamic**

(continued from last page)

mReactionsLocalParamSymbolsMaps

protected java.util.HashMap **mReactionsLocalParamSymbolsMaps**

mReactionRates

protected org.systemsbiology.math.Value **mReactionRates**

mReactionsDelayedReactionAssociations

protected org.systemsbiology.chem.DelayedReactionSolver
mReactionsDelayedReactionAssociations

mReactionSymbols

protected org.systemsbiology.math.Symbol **mReactionSymbols**

Constructors

Simulator

public **Simulator**()

Methods

hasDelayedReactionSolvers

protected boolean **hasDelayedReactionSolvers**()

isInitialized

public final boolean **isInitialized**()

canComputeFluctuations

public boolean **canComputeFluctuations**()

resizeDelayedReactionSolvers

protected final void **resizeDelayedReactionSolvers**(int pNumHistoryBins)

prepareForSimulation

protected final void **prepareForSimulation**(double pStartTime)

(continued from last page)

isStochasticSimulator

```
public abstract boolean isStochasticSimulator()
```

initializeSimulator

```
protected final void initializeSimulator(Model pModel)  
    throws DataNotFoundException
```

setInitialized

```
protected void setInitialized(boolean pInitialized)
```

clearSimulatorState

```
protected void clearSimulatorState()
```

initializeDynamicSymbolAdjustmentVectors

```
protected final void initializeDynamicSymbolAdjustmentVectors()
```

getAlias

```
public abstract java.lang.String getAlias()
```

createSimulationResults

```
protected final SimulationResults createSimulationResults(double pStartTime,  
    double pEndTime,  
    SimulatorParameters  
    pSimulatorParameters,  
    java.lang.String  
    pResultsSymbolNames,  
    double pResultsTimeValues,  
    java.lang.Object  
    pResultsSymbolValues,  
    double  
    pResultsFinalSymbolFluctuations)
```

addRequestedSymbolValues

```
protected final int addRequestedSymbolValues(double pCurTime,  
    int pLastTimeIndex,  
    Symbol pRequestedSymbols,  
    double pTimeValues,  
    java.lang.Object pRetSymbolValues)  
    throws DataNotFoundException
```

createTimesArray

```
protected final static double[] createTimesArray(double pStartTime,  
                                                double pEndTime,  
                                                int pNumTimePoints)
```

createRequestedSymbolArray

```
protected final Symbol[] createRequestedSymbolArray(java.util.HashMap pSymbolMap,  
                                                    java.lang.String  
pRequestedSymbols)  
                                                    throws DataNotFoundException
```

checkSimulationParameters

```
public final void checkSimulationParameters(double pStartTime,  
                                             double pEndTime,  
                                             SimulatorParameters pSimulatorParameters,  
                                             int pNumResultsTimePoints)  
                                             throws java.lang.IllegalArgumentException,  
                                             java.lang.IllegalStateException
```

checkSimulationParametersImpl

```
protected abstract void checkSimulationParametersImpl(SimulatorParameters  
pSimulatorParameters,  
                                                       int pNumResultsTimePoints)
```

computeReactionProbabilities

```
protected final void computeReactionProbabilities()  
                    throws DataNotFoundException
```

getDelayedReactionEstimatedAverageFutureRate

```
protected final static double getDelayedReactionEstimatedAverageFutureRate  
(SymbolEvaluatorChem pSymbolEvaluator,  
DelayedReactionSolver pDelayedReactionSolvers)  
DataNotFoundException  
                                                    throws
```

clearExpressionValueCaches

```
protected final void clearExpressionValueCaches()
```

computeDerivative

```
protected final void computeDerivative(double pTempDynamicSymbolValues,  
                                         double pDynamicSymbolDerivatives)  
                                         throws DataNotFoundException
```

(continued from last page)

setProgressReporter

```
public final void setProgressReporter(SimulationProgressReporter  
pSimulationProgressReporter)
```

setController

```
public final void setController(SimulationController pSimulationController)
```

setStatusUpdateIntervalSeconds

```
public final void setStatusUpdateIntervalSeconds(double pUpdateIntervalSeconds)  
throws java.lang.IllegalArgumentException
```

setMinNumMillisecondsForUpdate

```
protected final void setMinNumMillisecondsForUpdate(long pMinNumMillisecondsForUpdate)
```

getMinDelayedReactionDelay

```
protected final double getMinDelayedReactionDelay()
```

computeRatePartialDerivativeExpression

```
protected Expression computeRatePartialDerivativeExpression(int pReactionCtr,  
Expression  
pRateExpression,  
SymbolValue pSymbolValue,  
SymbolEvaluatorChem  
pSymbolEvaluator)  
throws DataNotFoundException
```

getReactionRateExpressions

```
public static Expression\[\] getReactionRateExpressions(Reaction pReactions)  
throws DataNotFoundException
```

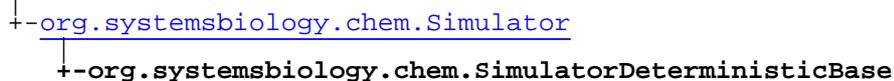
checkSimulationParametersForDeterministicSimulator

```
protected void checkSimulationParametersForDeterministicSimulator(SimulatorParameters  
pSimulatorParameters,  
int  
pNumResultsTimePoints)
```

org.systemsbiology.chem

Class SimulatorDeterministicBase

java.lang.Object



Direct Known Subclasses:

[SimulatorDeterministicRungeKuttaFixed](#), [SimulatorDeterministicRungeKuttaAdaptive](#)

```
public abstract class SimulatorDeterministicBase
extends Simulator
```

Simulates the dynamics of a set of coupled chemical reactions described by { @link Reaction } objects using the Runge-Kutta algorithm (fifth order with adaptive step-size control).

Fields

DEFAULT_MAX_ALLOWED_RELATIVE_ERROR

```
public static final double DEFAULT_MAX_ALLOWED_RELATIVE_ERROR
```

DEFAULT_MAX_ALLOWED_ABSOLUTE_ERROR

```
public static final double DEFAULT_MAX_ALLOWED_ABSOLUTE_ERROR
```

DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIONS

```
public static final boolean DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIONS
```

DEFAULT_NUM_HISTORY_BINS

```
protected static final int DEFAULT_NUM_HISTORY_BINS
```

mRKScratchPad

```
protected org.systemsbiology.chem.SimulatorDeterministicBase.RKScratchPad
mRKScratchPad
```

Constructors

SimulatorDeterministicBase

```
public SimulatorDeterministicBase()
```

Methods

iterate

```
protected abstract double iterate(double pNewDynamicSymbolValues)
    throws DataNotFoundException,
    AccuracyException
```

rk4step

```
protected final void rk4step(double pTimeStepSize,
    double pNewDynamicSymbolValues)
    throws DataNotFoundException
```

initialize

```
public void initialize(Model pModel)
    throws DataNotFoundException
```

computeScale

```
protected final void computeScale(double pTimeStepSize,
    double yscale)
    throws DataNotFoundException,
    AccuracyException
```

rkqc

```
protected final void rkqc(double pTimeStepSize,
    double pDynamicSymbolValueScales,
    double pNewDynamicSymbolValues,
    MutableDouble pRetAggregateRelativeError,
    MutableDouble pRetAggregateAbsoluteError)
    throws DataNotFoundException
```

setupErrorTolerances

```
protected abstract void setupErrorTolerances(SimulatorParameters pSimulatorParams,
    SimulatorDeterministicBase.RKScratchPad
    pRKScratchPad)
```

simulate

```
public SimulationResults simulate(double pStartTime,
    double pEndTime,
    SimulatorParameters pSimulatorParameters,
    int pNumResultsTimePoints,
    java.lang.String pRequestedSymbolNames)
    throws DataNotFoundException,
    java.lang.IllegalStateException,
    AccuracyException
```

(continued from last page)

getDefaultSimulatorParameters

```
public SimulatorParameters getDefaultSimulatorParameters()
```

isStochasticSimulator

```
public boolean isStochasticSimulator()
```

allowsInterrupt

```
public boolean allowsInterrupt()
```

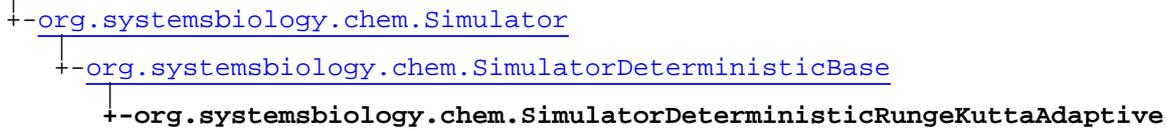
setupImpl

```
protected abstract void setupImpl(double pDeltaTime,  
                                   int pNumResultsTimePoints,  
                                   SimulatorParameters pSimulatorParams,  
                                   SimulatorDeterministicBase.RKScratchPad  
pRKScratchPad)
```

org.systemsbiology.chem

Class SimulatorDeterministicRungeKuttaAdaptive

java.lang.Object



All Implemented interfaces:

[ISimulator](#), [IAliasableClass](#)public final class **SimulatorDeterministicRungeKuttaAdaptive**extends [SimulatorDeterministicBase](#)implements [IAliasableClass](#), [ISimulator](#)

Simulates the dynamics of a set of coupled chemical reactions described by { @link Reaction } objects using the Runge-Kutta algorithm (fifth order with adaptive step-size control).

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

Constructors

SimulatorDeterministicRungeKuttaAdaptive

public **SimulatorDeterministicRungeKuttaAdaptive**()

Methods

iterate

```
protected double iterate(double pNewDynamicSymbolValues)
    throws DataNotFoundException,
           AccuracyException
```

setErrorTolerances

```
protected void setErrorTolerances(SimulatorParameters pSimulatorParams,
    SimulatorDeterministicBase.RKScratchPad
    pRKScratchPad)
```

(continued from last page)

setupImpl

```
protected void setupImpl(double pDeltaTime,  
                          int pNumResultsTimePoints,  
                          SimulatorParameters pSimulatorParams,  
                          SimulatorDeterministicBase.RKScratchPad pRKScratchPad)
```

checkSimulationParametersImpl

```
public void checkSimulationParametersImpl(SimulatorParameters pSimulatorParameters,  
                                           int pNumResultsTimePoints)
```

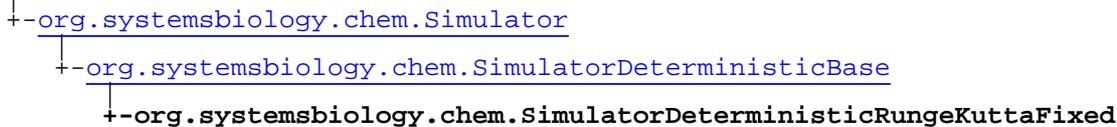
getAlias

```
public java.lang.String getAlias()
```

org.systemsbiology.chem

Class SimulatorDeterministicRungeKuttaFixed

java.lang.Object



All Implemented interfaces:

[ISimulator](#), [IAliasableClass](#)public final class **SimulatorDeterministicRungeKuttaFixed**extends [SimulatorDeterministicBase](#)implements [IAliasableClass](#), [ISimulator](#)

Simulates the dynamics of a set of coupled chemical reactions described by { @link Reaction } objects using the Runge-Kutta algorithm (fifth order with adaptive step-size control).

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

Constructors

SimulatorDeterministicRungeKuttaFixed

public **SimulatorDeterministicRungeKuttaFixed**()

Methods

setupErrorTolerances

```
protected void setupErrorTolerances(SimulatorParameters pSimulatorParams,
                                     SimulatorDeterministicBase.RKScratchPad
                                     pRKScratchPad)
```

iterate

```
protected double iterate(double pNewDynamicSymbolValues)
    throws DataNotFoundException,
           AccuracyException
```

(continued from last page)

getMaxStepSize

```
protected double getMaxStepSize(double pDeltaTime,  
                                long pNumResultsTimePoints,  
                                SimulatorParameters pSimulatorParams)
```

setupImpl

```
protected void setupImpl(double pDeltaTime,  
                          int pNumResultsTimePoints,  
                          SimulatorParameters pSimulatorParams,  
                          SimulatorDeterministicBase.RKScratchPad pRKScratchPad)
```

checkSimulationParametersImpl

```
public void checkSimulationParametersImpl(SimulatorParameters pSimulatorParameters,  
                                           int pNumResultsTimePoints)
```

getAlias

```
public java.lang.String getAlias()
```

org.systemsbiology.chem

Class SimulatorParameters

java.lang.Object

└-org.systemsbiology.chem.SimulatorParameters

public class **SimulatorParameters**extends java.lang.Object

Constructors

SimulatorParameters

public **SimulatorParameters**()

Methods

toString

public java.lang.String **toString**()

setStepSizeFraction

public void **setStepSizeFraction**(java.lang.Double pStepSizeFraction)

setStepSizeFraction

public void **setStepSizeFraction**(double pStepSizeFraction)

getStepSizeFraction

public java.lang.Double **getStepSizeFraction**()

setNumHistoryBins

public void **setNumHistoryBins**(int pNumHistoryBins)

getNumHistoryBins

public java.lang.Integer **getNumHistoryBins**()

(continued from last page)

setComputeFluctuations

```
public void setComputeFluctuations(boolean pComputeFluctuations)
```

getComputeFluctuations

```
public java.lang.Boolean getComputeFluctuations()
```

setEnsembleSize

```
public void setEnsembleSize(java.lang.Integer pEnsembleSize)
```

setEnsembleSize

```
public void setEnsembleSize(int pEnsembleSize)
```

getEnsembleSize

```
public java.lang.Integer getEnsembleSize()
```

setMaxAllowedRelativeError

```
public void setMaxAllowedRelativeError(java.lang.Double pMaxAllowedRelativeError)
```

setMaxAllowedRelativeError

```
public void setMaxAllowedRelativeError(double pMaxAllowedRelativeError)
```

getMaxAllowedRelativeError

```
public java.lang.Double getMaxAllowedRelativeError()
```

setMaxAllowedAbsoluteError

```
public void setMaxAllowedAbsoluteError(java.lang.Double pMaxAllowedAbsoluteError)
```

setMaxAllowedAbsoluteError

```
public void setMaxAllowedAbsoluteError(double pMaxAllowedAbsoluteError)
```

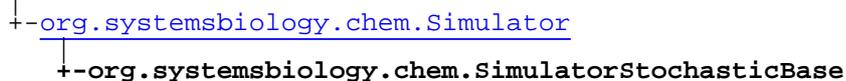
getMaxAllowedAbsoluteError

```
public java.lang.Double getMaxAllowedAbsoluteError()
```

org.systemsbiology.chem

Class SimulatorStochasticBase

java.lang.Object



Direct Known Subclasses:

[SimulatorStochasticGillespie](#), [SimulatorStochasticTauLeapBase](#),
[SimulatorStochasticGibsonBruck](#)

public abstract class **SimulatorStochasticBase**extends [Simulator](#)

Base class for all stochastic simulators.

Fields

DEFAULT_ENSEMBLE_SIZE

public static final int **DEFAULT_ENSEMBLE_SIZE**

DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIONS

public static final boolean **DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIONS**

DEFAULT_NUM_HISTORY_BINS

public static final int **DEFAULT_NUM_HISTORY_BINS**

mRandomNumberGenerator

protected edu.cornell.lasp.houle.RngPack.RandomElement **mRandomNumberGenerator**

mPoissonEventGenerator

protected cern.jet.random.Poisson **mPoissonEventGenerator**

mDynamicSymbolDelayedReactionAssociations

protected org.systemsbiology.chem.DelayedReactionSolver
mDynamicSymbolDelayedReactionAssociations

Constructors

(continued from last page)

SimulatorStochasticBase

```
public SimulatorStochasticBase()
```

Methods

modifyDefaultSimulatorParameters

```
protected abstract void modifyDefaultSimulatorParameters(SimulatorParameters  
pSimulatorParameters)
```

setRandomNumberGenerator

```
protected void setRandomNumberGenerator(edu.cornell.lassp.houle.RngPack.RandomElement  
pRandomNumberGenerator)
```

getRandomNumberGenerator

```
protected edu.cornell.lassp.houle.RngPack.RandomElement getRandomNumberGenerator()
```

getRandomNumberUniformInterval

```
protected final static double getRandomNumberUniformInterval  
(edu.cornell.lassp.houle.RngPack.RandomElement pRandomNumberGenerator)
```

initializeRandomNumberGenerator

```
protected void initializeRandomNumberGenerator()
```

initializePoissonEventGenerator

```
protected void initializePoissonEventGenerator()
```

checkDynamicalSymbolsValues

```
protected void checkDynamicalSymbolsValues(boolean pSimulationIsRunning,  
SymbolEvaluatorChem pSymbolEvaluator)  
throws AccuracyException
```

getNextDelayedReactionIndex

```
protected final static int getNextDelayedReactionIndex(DelayedReactionSolver  
pDelayedReactionSolvers)
```

(continued from last page)

updateSymbolValuesForReaction

```
protected final void updateSymbolValuesForReaction(int pReactionCtr,  
                                                    double pDynamicSymbolValues,  
                                                    DelayedReactionSolver  
pDynamicSymbolDelayedReactionAssociations,  
                                                    long pNumberFirings)  
throws DataNotFoundException
```

chooseDeltaTimeToNextReaction

```
protected final double chooseDeltaTimeToNextReaction(double pReactionProbability)
```

isStochasticSimulator

```
public boolean isStochasticSimulator()
```

iterate

```
protected abstract double iterate(MutableInteger pLastReactionIndex)  
throws DataNotFoundException,  
java.lang.IllegalStateException,  
AccuracyException
```

prepareForStochasticSimulation

```
protected abstract void prepareForStochasticSimulation(double pStartTime,  
                                                       SimulatorParameters  
pSimulatorParameters)  
throws DataNotFoundException,  
java.lang.IllegalArgumentException
```

getPoissonEvent

```
protected final static int getPoissonEvent(cern.jet.random.Poisson  
pPoissonEventGenerator,  
double pMean)
```

initializeSimulatorStochastic

```
protected final void initializeSimulatorStochastic(Model pModel)  
throws InvalidInputException
```

chooseIndexOfNextReaction

```
protected final int chooseIndexOfNextReaction(double  
pAggregateReactionProbabilityDensity)  
throws java.lang.IllegalArgumentException
```

(continued from last page)

simulate

```
public final SimulationResults simulate(double pStartTime,  
                                           double pEndTime,  
                                           SimulatorParameters pSimulatorParameters,  
                                           int pNumResultsTimePoints,  
                                           java.lang.String pRequestedSymbolNames)  
    throws DataNotFoundException,  
           java.lang.IllegalStateException,  
           java.lang.IllegalArgumentException,  
           AccuracyException
```

allowsInterrupt

```
public boolean allowsInterrupt()
```

checkSimulationParametersImpl

```
protected void checkSimulationParametersImpl(SimulatorParameters pSimulatorParameters,  
                                               int pNumResultsTimePoints)
```

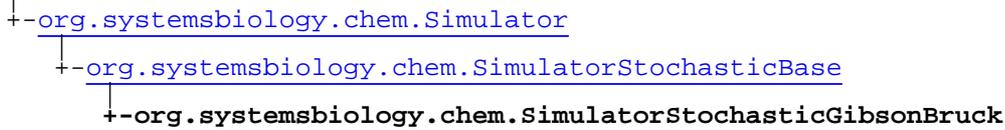
getDefaultSimulatorParameters

```
public SimulatorParameters getDefaultSimulatorParameters()
```

org.systemsbiology.chem

Class SimulatorStochasticGibsonBruck

java.lang.Object



All Implemented interfaces:

[ISimulator](#), [IAliasableClass](#)public final class **SimulatorStochasticGibsonBruck**extends [SimulatorStochasticBase](#)implements [IAliasableClass](#), [ISimulator](#)

Implementation of the Gibson-Bruck "Next Reaction" algorithm. Note: If the Model contains M reactions with custom reaction rate expressions, the Gibson algorithm will require M^2 memory because each of the M reactions will need to be recomputed when any of the other reactions occurs.

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

Constructors

SimulatorStochasticGibsonBruck

public **simulatorStochasticGibsonBruck**()

Methods

prepareForStochasticSimulation

```
protected void prepareForStochasticSimulation(double pStartTime,
                                             SimulatorParameters
pSimulatorParameters)
                                             throws DataNotFoundException
```

iterate

```
protected double iterate(MutableInteger pLastReactionIndex)
                        throws DataNotFoundException,
                        java.lang.IllegalStateException
```

(continued from last page)

initialize

```
public void initialize(Model pModel)
    throws DataNotFoundException,
           InvalidInputException
```

modifyDefaultSimulatorParameters

```
protected void modifyDefaultSimulatorParameters(SimulatorParameters
pSimulatorParameters)
```

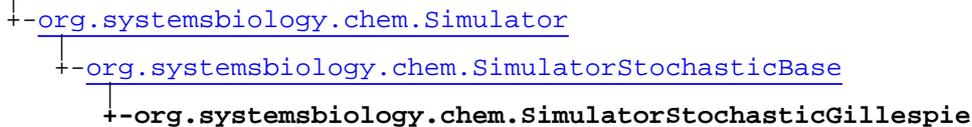
getAlias

```
public java.lang.String getAlias()
```

org.systemsbiology.chem

Class SimulatorStochasticGillespie

java.lang.Object



All Implemented interfaces:

[ISimulator](#), [IAliasableClass](#)

```

public final class SimulatorStochasticGillespie
extends SimulatorStochasticBase
implements IAliasableClass, ISimulator
  
```

Simulates the dynamics of a set of coupled chemical reactions described by { @link Reaction } objects using the Gillespie stochastic algorithm, "direct method".

Fields

CLASS_ALIAS

```
public static final java.lang.String CLASS_ALIAS
```

Constructors

SimulatorStochasticGillespie

```
public SimulatorStochasticGillespie()
```

Methods

prepareForStochasticSimulation

```
protected void prepareForStochasticSimulation(double pStartTime,
                                               SimulatorParameters
                                               pSimulatorParameters)
```

iterate

```
protected double iterate(MutableInteger pLastReactionIndex)
    throws DataNotFoundException,
           java.lang.IllegalStateException
```

(continued from last page)

initialize

```
public void initialize(Model pModel)
    throws DataNotFoundException,
           InvalidInputException
```

modifyDefaultSimulatorParameters

```
protected void modifyDefaultSimulatorParameters(SimulatorParameters
pSimulatorParameters)
```

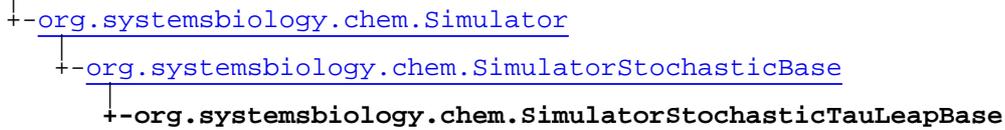
getAlias

```
public java.lang.String getAlias()
```

org.systemsbiology.chem

Class SimulatorStochasticTauLeapBase

java.lang.Object



Direct Known Subclasses:

[SimulatorStochasticTauLeapSimple](#), [SimulatorStochasticTauLeapComplex](#)public abstract class **SimulatorStochasticTauLeapBase**extends [SimulatorStochasticBase](#)

Simulates the dynamics of a set of coupled chemical reactions described by { @link Reaction } objects using the Gillespie stochastic algorithm, "tau-leap method".

Fields

mReactionHasLocalSymbolsFlags

protected boolean **mReactionHasLocalSymbolsFlags**

mAllowedError

protected double **mAllowedError**

mNumNonLeapIterationsSinceLastLeapCheck

protected long **mNumNonLeapIterationsSinceLastLeapCheck**

mLastIterationWasLeap

protected boolean **mLastIterationWasLeap**

mMinRatioOfLeapTimeToReactionTimeScale

protected long **mMinRatioOfLeapTimeToReactionTimeScale**

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

(continued from last page)

Constructors

SimulatorStochasticTauLeapBase

```
public SimulatorStochasticTauLeapBase()
```

Methods

prepareForStochasticSimulation

```
protected void prepareForStochasticSimulation(double pStartTime,  
                                              SimulatorParameters  
pSimulatorParameters)  
                                              throws java.lang.IllegalArgumentException
```

iterate

```
protected double iterate(MutableInteger pLastReactionIndex)  
                       throws DataNotFoundException,  
                               java.lang.IllegalStateException,  
                               AccuracyException
```

computeLeapTime

```
protected abstract double computeLeapTime(double pSumReactionProbabilities)  
                               throws DataNotFoundException
```

initializeTauLeap

```
protected abstract void initializeTauLeap(SymbolEvaluatorChem pSymbolEvaluator)  
                               throws DataNotFoundException,  
                                       InvalidInputException
```

initialize

```
public void initialize(Model pModel)  
           throws DataNotFoundException,  
                   InvalidInputException
```

getAlias

```
public java.lang.String getAlias()
```

modifyDefaultSimulatorParameters

```
protected void modifyDefaultSimulatorParameters(SimulatorParameters  
pSimulatorParameters)
```

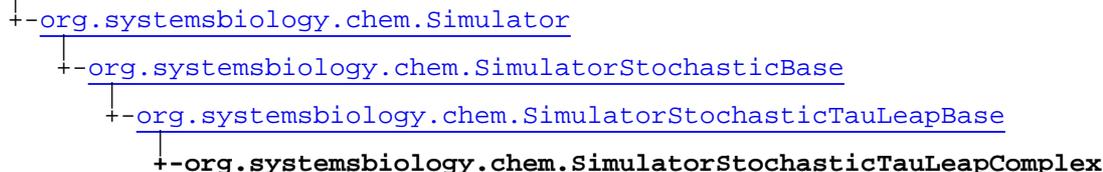
checkSimulationParametersImpl

```
protected void checkSimulationParametersImpl(SimulatorParameters pSimulatorParameters,  
                                              int pNumResultsTimePoints)
```

org.systemsbiology.chem

Class SimulatorStochasticTauLeapComplex

java.lang.Object



All Implemented interfaces:

[ISimulator](#), [IAliasableClass](#)public class **SimulatorStochasticTauLeapComplex**extends [SimulatorStochasticTauLeapBase](#)implements [IAliasableClass](#), [ISimulator](#)

Implementation of Gillespie's "Tau-Leap" simulator that is optimized for complex models (models with complex rate law expressions that will have very complicated partial derivatives).

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

Constructors

SimulatorStochasticTauLeapComplex

public **SimulatorStochasticTauLeapComplex**()

Methods

computeLeapTime

protected double **computeLeapTime**(double pSumReactionProbabilities)
throws DataNotFoundException

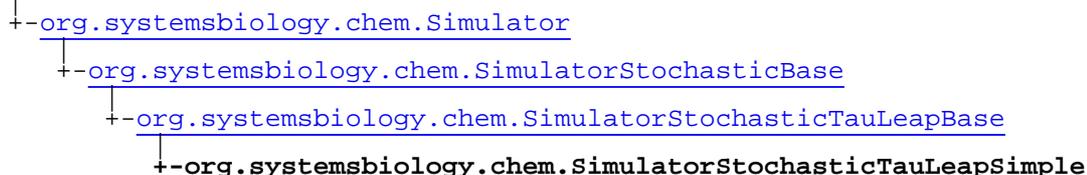
initializeTauLeap

protected void **initializeTauLeap**(SymbolEvaluatorChem pSymbolEvaluator)
throws DataNotFoundException

org.systemsbiology.chem

Class SimulatorStochasticTauLeapSimple

java.lang.Object



All Implemented interfaces:

[ISimulator](#), [IAliasableClass](#)

```

public class SimulatorStochasticTauLeapSimple
  extends SimulatorStochasticTauLeapBase
  implements IAliasableClass, ISimulator
  
```

Implementation of Gillespie's "Tau-Leap" simulator that is optimized for simple models (models with simple elementary reactions, rather than complex rate law expressions).

Fields

CLASS_ALIAS

```
public static final java.lang.String CLASS_ALIAS
```

Constructors

SimulatorStochasticTauLeapSimple

```
public SimulatorStochasticTauLeapSimple()
```

Methods

computeLeapTime

```
protected double computeLeapTime(double pSumReactionProbabilities)
  throws DataNotFoundException
```

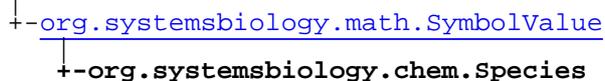
initializeTauLeap

```
protected void initializeTauLeap(SymbolEvaluatorChem pSymbolEvaluator)
  throws DataNotFoundException,
  InvalidInputException
```

org.systemsbiology.chem

Class Species

java.lang.Object

public final class **Species**extends [SymbolValue](#)

Represents a distinct, named chemical species. Must reside in a {@link Compartment}. The species name does not have to be globally unique; two species can have the same name, if they reside in a different compartment. The "symbol name" is constructed using the species name and the compartment name, and it will be different between the two species. A species object has a value associated with it, by virtue of the superclass {@link org.systemsbiology.math.SymbolValue}. This value represents the amount of the species present, in molecules.

Constructors

Species

```
public Species(java.lang.String pName,
                Compartment pCompartment)
```

Species

```
public Species(SymbolValue pSymbolValue,
                Compartment pCompartment)
```

Methods

setSpeciesPopulation

```
public void setSpeciesPopulation(double pSpeciesPopulation)
```

setSpeciesPopulation

```
public void setSpeciesPopulation(Expression pSpeciesPopulation)
```

getName

```
public java.lang.String getName()
```

getCompartment

```
public Compartment getCompartment()
```

equals

```
public boolean equals(Species pSpecies)
```

clone

```
public java.lang.Object clone()
```

toString

```
public java.lang.String toString()
```

org.systemsbiology.chem

Class SteadyStateAnalyzer

java.lang.Object

```

  |
  +--org.systemsbiology.chem.SteadyStateAnalyzer

```

```

public final class SteadyStateAnalyzer
extends java.lang.Object

```

Class providing functions for analyzing the steady-state behavior of a {@link Model}. To use this class, you must have previously run a simulation using a {@link ISimulator}, and have references to the {@link SymbolEvaluatorChem} object that will return the final values for species concentrations in the model, resulting from the simulation.

Constructors

SteadyStateAnalyzer

```

public SteadyStateAnalyzer(Model pModel)

```

Methods

computeJacobian

```

public static java.lang.Object[] computeJacobian(Expression pReactionRateExpressions,
                                                  Reaction pReactions,
                                                  Species pSpecies,
                                                  java.lang.Object
pReactionSpeciesAdjustmentVectors,
                                                  SymbolEvaluatorChem pSymbolEvaluator)
throws DataNotFoundException

```

estimateSpeciesFluctuations

```

public static double[] estimateSpeciesFluctuations(Reaction pReactions,
                                                  Species pSpecies,
                                                  java.lang.Object
pReactionSpeciesAdjustmentVectors,
                                                  double pReactionProbabilities,
                                                  SymbolEvaluatorChem
pSymbolEvaluator)
throws DataNotFoundException

```

For the supplied reactions and species, estimates the steady-state species fluctuations in the model. If the estimate is unsuccessful (e.g., because of the absence of an eigenvalue with negative real part), null is returned.

org.systemsbiology.chem

Class SymbolEvaluatorChem

```
java.lang.Object
  |
  +- org.systemsbiology.math.SymbolEvaluator
    |
    +- org.systemsbiology.chem.SymbolEvaluatorChem
```

```
public final class SymbolEvaluatorChem
extends SymbolEvaluator
```

Symbol Evaluator class used for chemical simulations.

Constructors

SymbolEvaluatorChem

```
public SymbolEvaluatorChem(boolean pUseExpressionValueCaching,
                           SymbolEvaluationPostProcessor
                           pSymbolEvaluationPostProcessor,
                           ReservedSymbolMapper pReservedSymbolMapper,
                           java.util.HashMap pSymbolsMap)
```

Methods

getReservedSymbolMapper

```
public ReservedSymbolMapper getReservedSymbolMapper()
```

setLocalSymbolsMap

```
public void setLocalSymbolsMap(java.util.HashMap pLocalSymbolsMap)
```

setTime

```
public void setTime(double pTime)
```

getTime

```
public double getTime()
```

getExpressionValue

```
public Expression getExpressionValue(Symbol pSymbol)
                           throws DataNotFoundException
```

getUnindexedValue

```
public double getUnindexedValue(Symbol pSymbol)
    throws DataNotFoundException,
        java.lang.IllegalStateException
```

hasValue

```
public boolean hasValue(Symbol pSymbol)
```

getSymbol

```
public Symbol getSymbol(java.lang.String pSymbolName)
```

org.systemsbiochemistrychem

Class TimeSeriesOutputFormat

```
java.lang.Object
  |
  +-org.systemsbiochemistrychem.TimeSeriesOutputFormat
```

```
public class TimeSeriesOutputFormat
  extends java.lang.Object
```

Enumeration class for output file formats that are supported by the {@link TimeSeriesSymbolValuesReporter} class.

Fields

CSV_EXCEL

```
public static final org.systemsbiochemistrychem.TimeSeriesOutputFormat CSV_EXCEL
```

CSV_MATLAB

```
public static final org.systemsbiochemistrychem.TimeSeriesOutputFormat CSV_MATLAB
```

CSV_GNUPLOT

```
public static final org.systemsbiochemistrychem.TimeSeriesOutputFormat CSV_GNUPLOT
```

Methods

toString

```
public java.lang.String toString()
```

get

```
public static TimeSeriesOutputFormat get(java.lang.String pName)
```

updateDecimalFormatSymbols

```
public void updateDecimalFormatSymbols(java.text.DecimalFormatSymbols
  pDecimalFormatSymbols)
```

getNaN

```
public java.lang.String getNaN()
```

getInfinity

```
public java.lang.String getInfinity()
```

getSortedFileFormatNames

```
public static java.lang.String[] getSortedFileFormatNames()
```

getCommentChar

```
public char getCommentChar()
```

org.systemsbiology.chem

Class TimeSeriesSymbolValuesReporter

java.lang.Object

└-org.systemsbiology.chem.TimeSeriesSymbolValuesReporter

```
public class TimeSeriesSymbolValuesReporter
extends java.lang.Object
```

Class for printing time-series data to a PrintWriter.

Constructors

TimeSeriesSymbolValuesReporter

```
public TimeSeriesSymbolValuesReporter()
```

Methods

reportTimeSeriesSymbolValues

```
public final static void reportTimeSeriesSymbolValues(java.io.PrintWriter
pPrintWriter,
                                                    java.lang.String
pRequestedSymbolNames,
                                                    double pTimeValues,
                                                    java.lang.Object pSymbolValues,
                                                    TimeSeriesOutputFormat
pTimeSeriesOutputFormat)
                                                    throws
java.lang.IllegalArgumentException
```

reportTimeSeriesSymbolValues

```
public final static void reportTimeSeriesSymbolValues(java.io.PrintWriter
pPrintWriter,
                                                    java.lang.String
pRequestedSymbolNames,
                                                    double pTimeValues,
                                                    java.lang.Object pSymbolValues,
                                                    java.text.NumberFormat
pNumberFormat,
                                                    TimeSeriesOutputFormat
pTimeSeriesOutputFormat)
                                                    throws
java.lang.IllegalArgumentException
```

Package

org.systemsbiology.chem.sbw

This package contains classes used to provide an interface to the `org.systemsbiology.chem` package via the Systems Biology Workbench (SBW) system. The [overview for the SBW system is available on the web](#). </p>

In order to compile this package, you will need to have the SBW system installed on the build system and the target (runtime) system. In particular, this package makes use of the JAR files

```
/usr/local/lib/SBWCore.jar  
/usr/local/modules/SBWValidate.jar  
whre &quot;/usr/local  
p>
```

org.systemsbiology.chem.sbw

Interface **ISimulationDriverService**

public interface **ISimulationDriverService**

Methods

doAnalysis

```
public void doAnalysis(java.lang.String pModelDefinitionText)
```

org.systemsbiology.chem.sbw

Class **SimulationDriverModule**

java.lang.Object

└-org.systemsbiology.chem.sbw.SimulationDriverModule

public class **SimulationDriverModule**

extends java.lang.Object

Fields

MODULE_UNIQUE_NAME

public static final java.lang.String **MODULE_UNIQUE_NAME**

MODULE_DISPLAY_NAME

public static final java.lang.String **MODULE_DISPLAY_NAME**

CATEGORY_NAME

public static final java.lang.String **CATEGORY_NAME**

Constructors

SimulationDriverModule

public **SimulationDriverModule**()

Methods

getConfig

public static [AppConfig](#) **getConfig**()

main

public static void **main**(java.lang.String pArgs)

org.systemsbiology.chem.sbw

Class SimulationDriverService

java.lang.Object

└--org.systemsbiology.chem.sbw.SimulationDriverService

All Implemented interfaces:

[ISimulationDriverService](#)

```
public class SimulationDriverService
  extends java.lang.Object
  implements ISimulationDriverService
```

Fields

SERVICE_NAME

```
public static final java.lang.String SERVICE_NAME
```

SERVICE_DESCRIPTION

```
public static final java.lang.String SERVICE_DESCRIPTION
```

Constructors

SimulationDriverService

```
public SimulationDriverService()
```

Methods

doAnalysis

```
public void doAnalysis(java.lang.String pModelDescriptionText)
```

doAnalysisCMDL

```
public void doAnalysisCMDL(java.lang.String pModelDescriptionText)
```

Package

org.systemsbiology.chem.sbml

p>

org.systemsbiology.chem.sbml

Class MarkupLanguageImporter

java.lang.Object

└-org.systemsbiology.chem.sbml.MarkupLanguageImporter

```
public class MarkupLanguageImporter
extends java.lang.Object
```

Proxy class for interrogating a parsed SBML document. Uses the SBMLValidate.SBMLReader class to parse and query an SBML document contained in a String. Used by the {@link ModelBuilderMarkupLanguage} class.

See Also:

[org.systemsbiology.chem.sbml.ModelBuilderMarkupLanguage](#)

Constructors

MarkupLanguageImporter

```
public MarkupLanguageImporter()
```

Methods

readModelDescription

```
public void readModelDescription(java.lang.String pModelDescription)
throws InvalidInputException
```

getModelName

```
public java.lang.String getModelName()
```

getNumCompartments

```
public int getNumCompartments()
```

getNumReactions

```
public int getNumReactions()
```

getNumFloatingSpecies

```
public int getNumFloatingSpecies()
```

(continued from last page)

getNumBoundarySpecies

```
public int getNumBoundarySpecies()
```

getNumGlobalParameters

```
public int getNumGlobalParameters()
```

getNthCompartmentName

```
public java.lang.String getNthCompartmentName(int compartment)
```

getNthFloatingSpeciesName

```
public java.lang.String getNthFloatingSpeciesName(int floatingSpecies)
```

getNthBoundarySpeciesName

```
public java.lang.String getNthBoundarySpeciesName(int boundarySpecies)
```

getNthFloatingSpeciesCompartmentName

```
public java.lang.String getNthFloatingSpeciesCompartmentName(int floatingSpecies)
```

getNthBoundarySpeciesCompartmentName

```
public java.lang.String getNthBoundarySpeciesCompartmentName(int boundarySpecies)
```

getNthReactionName

```
public java.lang.String getNthReactionName(int reaction)
```

getNumReactants

```
public int getNumReactants(int reaction)
```

getNumProducts

```
public int getNumProducts(int reaction)
```

getNthReactantName

```
public java.lang.String getNthReactantName(int reaction,  
                                             int reactant)
```

(continued from last page)

getNthProductName

```
public java.lang.String getNthProductName(int reaction,  
                                           int product)
```

getKineticLaw

```
public java.lang.String getKineticLaw(int reaction)
```

getNthReactantStoichiometry

```
public int getNthReactantStoichiometry(int reaction,  
                                         int reactant)
```

getNthProductStoichiometry

```
public int getNthProductStoichiometry(int reaction,  
                                         int product)
```

getNumParameters

```
public int getNumParameters(int reaction)
```

getNthParameterName

```
public java.lang.String getNthParameterName(int reaction,  
                                              int parameter)
```

getNthParameterValue

```
public double getNthParameterValue(int reaction,  
                                     int parameter)
```

getNthParameterHasValue

```
public boolean getNthParameterHasValue(int reaction,  
                                         int parameter)
```

getNthGlobalParameterName

```
public java.lang.String getNthGlobalParameterName(int globalParameter)
```

(continued from last page)

hasValue

```
public boolean hasValue(java.lang.String name)
    throws InvalidInputException
```

getValue

```
public double getValue(java.lang.String name)
    throws InvalidInputException
```

getBuiltinFunctionInfo

```
public java.lang.String[] getBuiltinFunctionInfo(java.lang.String name)
```

getBuiltinFunctions

```
public java.lang.String[] getBuiltinFunctions()
```

getNumRules

```
public int getNumRules()
```

getNthRuleName

```
public java.lang.String getNthRuleName(int pIndex)
```

getNthRuleFormula

```
public java.lang.String getNthRuleFormula(int pIndex)
```

getNthRuleType

```
public java.lang.String getNthRuleType(int pIndex)
```

getSubstanceUnitsString

```
public java.lang.String getSubstanceUnitsString()
```

org.systemsbiology.chem.sbml

Class ModelBuilderMarkupLanguage

java.lang.Object

└-org.systemsbiology.chem.sbml.ModelBuilderMarkupLanguage

All Implemented interfaces:

[IAliasableClass](#), [IModelBuilder](#)

```
public class ModelBuilderMarkupLanguage
extends java.lang.Object
implements IModelBuilder, IAliasableClass
```

Builds a { @link org.systemsbiology.chem.Model model } from SBML input. The SBML must be valid SBML level 1 (version 1 or 2), with the following restrictions:

- units are not allowed
- rule definitions of type "rate" are not allowed
- pre-defined kinetic law functions such as "uui" and "massi" are not supported

Fields

CLASS_ALIAS

```
public static final java.lang.String CLASS_ALIAS
```

Constructors

ModelBuilderMarkupLanguage

```
public ModelBuilderMarkupLanguage()
```

Methods

readModel

```
public java.lang.String readModel(java.io.InputStream pInputStream)
    throws InvalidInputException
```

writeModel

```
public void writeModel(java.lang.String pOutputData,
    java.io.OutputStream pOutputStream)
    throws InvalidInputException
```

(continued from last page)

buildModel

```
public Model buildModel(java.io.InputStream pInputStream,  
                        IncludeHandler pIncludeHandler)  
    throws InvalidInputException,  
           java.io.IOException
```

Processes an SBML model into a {@link Model} object, and returns the model name. Note that according to the SBML specification, there can be only one model per SBML document or data stream.

getFileRegex

```
public java.lang.String getFileRegex()
```

org.systemsbiology.chem.sbml

Class ModelExporterMarkupLanguage

java.lang.Object

└-org.systemsbiology.chem.sbml.ModelExporterMarkupLanguage

All Implemented interfaces:

[IAliasableClass](#), [IModelExporter](#)

public class **ModelExporterMarkupLanguage**

extends java.lang.Object

implements [IModelExporter](#), [IAliasableClass](#)

Exports a {@link org.systemsbiology.chem.Model} to the Systems Biology Markup Language (SBML) format, Level 1, Version 1 or 2 (Version 2 is the default).

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

DEFAULT_REACTION_PARAMETER_SYMBOL_NAME

public static final java.lang.String **DEFAULT_REACTION_PARAMETER_SYMBOL_NAME**

DEFAULT_SPECIFICATION

public static final
org.systemsbiology.chem.sbml.ModelExporterMarkupLanguage.Specification
DEFAULT_SPECIFICATION

Constructors

ModelExporterMarkupLanguage

public **ModelExporterMarkupLanguage**()

Methods

(continued from last page)

export

```
public void export(Model pModel,  
                  java.io.PrintWriter pOutputWriter)  
    throws java.lang.IllegalArgumentException,  
           DataNotFoundException,  
           java.lang.IllegalStateException,  
           java.lang.UnsupportedOperationException,  
           ModelExporterException
```

export

```
public void export(Model pModel,  
                  java.io.PrintWriter pOutputWriter,  
                  ModelExporterMarkupLanguage.Specification pSpecification)  
    throws java.lang.IllegalArgumentException,  
           DataNotFoundException,  
           java.lang.IllegalStateException,  
           java.lang.UnsupportedOperationException,  
           ModelExporterException
```

Given a {[@link org.systemsbiology.chem.Model](#)} object defining a system of chemical reactions and the initial species populations, writes out the SBML description of the model and initial species populations, to the output stream contained in pOutputWriter.

getFileRegex

```
public java.lang.String getFileRegex()
```

org.systemsbiology.chem.sbml

Class ModelExporterMarkupLanguage.Specification

java.lang.Object

└-org.systemsbiology.chem.sbml.ModelExporterMarkupLanguage.Specification

public static class **ModelExporterMarkupLanguage.Specification**

extends java.lang.Object

Fields

LEVEL1_VERSION1

public static final

org.systemsbiology.chem.sbml.ModelExporterMarkupLanguage.Specification **LEVEL1_VERSION1**

LEVEL1_VERSION2

public static final

org.systemsbiology.chem.sbml.ModelExporterMarkupLanguage.Specification **LEVEL1_VERSION2**

Methods

getLevel

public java.lang.String **getLevel()**

getVersion

public java.lang.String **getVersion()**

toString

public java.lang.String **toString()**

org.systemsbiology.chem.sbml
Class SubstanceUnit

```
java.lang.Object  
  |  
  +--org.systemsbiology.chem.sbml.SubstanceUnit
```

```
public class SubstanceUnit  
extends java.lang.Object
```

Enumeration of all allowed substance units, for SBML models to be imported into a {@link org.systemsbiology.chem.Model}.

Fields

ITEM

```
public static final org.systemsbiology.chem.sbml.SubstanceUnit ITEM
```

MOLE

```
public static final org.systemsbiology.chem.sbml.SubstanceUnit MOLE
```

Methods

get

```
public static SubstanceUnit get(java.lang.String pName)
```

toString

```
public java.lang.String toString()
```

getConversionToMolecules

```
public double getConversionToMolecules()
```

org.systemsbiology.chem.sbml

Class SymbolEvaluationPostProcessorChemMarkupLanguage

java.lang.Object



```
public final class SymbolEvaluationPostProcessorChemMarkupLanguage
extends SymbolEvaluationPostProcessor
```

Constructors

SymbolEvaluationPostProcessorChemMarkupLanguage

```
public SymbolEvaluationPostProcessorChemMarkupLanguage( java.util.HashMap
pSpeciesCompartmentMap,
                                                    java.util.HashSet pReactionSet,
                                                    double
pConvertSubstanceToMolecules)
```

Methods

clone

```
public java.lang.Object clone()
```

getConvertSubstanceToMolecules

```
public double getConvertSubstanceToMolecules()
```

modifySymbol

```
public java.lang.String modifySymbol(Symbol pSymbol)
throws DataNotFoundException
```

modifyResult

```
public double modifyResult(Symbol pSymbol,
SymbolEvaluator pSymbolEvaluator,
double pSymbolValue)
throws DataNotFoundException
```

Package

org.systemsbiology.chem.odetojava

p>

org.systemsbiology.chem.odetojava

Class SimulatorOdeToJavaBase

java.lang.Object

├─ [org.systemsbiology.chem.Simulator](#)

└─ org.systemsbiology.chem.odetojava.SimulatorOdeToJavaBase

All Implemented interfaces:

odeToJava.modules.ODERecorder, odeToJava.modules.ODE

Direct Known Subclasses:[SimulatorOdeToJavaRungeKuttaAdaptive](#), [SimulatorOdeToJavaRungeKuttaImplicit](#)public abstract class **SimulatorOdeToJavaBase**extends [Simulator](#)

implements odeToJava.modules.ODE, odeToJava.modules.ODERecorder

Fields

DEFAULT_MAX_ALLOWED_RELATIVE_ERROR

public static final double DEFAULT_MAX_ALLOWED_RELATIVE_ERROR

DEFAULT_MAX_ALLOWED_ABSOLUTE_ERROR

public static final double DEFAULT_MAX_ALLOWED_ABSOLUTE_ERROR

DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIONS

public static final boolean DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIONS

DEFAULT_NUM_HISTORY_BINS

protected static final int DEFAULT_NUM_HISTORY_BINS

Constructors

SimulatorOdeToJavaBase

public SimulatorOdeToJavaBase()

Methods

(continued from last page)

runExternalSimulation

```
protected abstract void runExternalSimulation(odeToJava.modules.Span
pSimulationTimeSpan,
                                           double pInitialDynamicSymbolValues,
                                           double pInitialStepSize,
                                           double pMaxAllowedRelativeError,
                                           double pMaxAllowedAbsoluteError,
                                           java.lang.String pTempOutputFileName)
```

simulate

```
public SimulationResults simulate(double pStartTime,
                                   double pEndTime,
                                   SimulatorParameters pSimulatorParameters,
                                   int pNumResultsTimePoints,
                                   java.lang.String pRequestedSymbolNames)
    throws DataNotFoundException,
           java.lang.IllegalStateException,
           java.lang.IllegalArgumentException,
           AccuracyException,
           SimulationFailedException
```

readSimulationOutput

```
protected void readSimulationOutput(java.io.File pSimulationResultsFile,
                                     SymbolEvaluatorChem pSymbolEvaluator,
                                     double pDynamicSymbolValues,
                                     double pStartTime,
                                     double pEndTime,
                                     int pNumResultsTimePoints,
                                     Symbol pRequestedSymbols,
                                     double pRetResultsTimeValues,
                                     java.lang.Object pRetResultsSymbolValues)
    throws InvalidInputException,
           DataNotFoundException,
           java.io.FileNotFoundException,
           java.io.IOException,
           SimulationFailedException
```

f

```
public final double[] f(double t,
                        double x)
```

g

```
public final double[] g(double t,
                        double x)
```

record

```
public final void record(double t,
                        double x)
```

initialize

```
public void initialize(Model pModel)
    throws DataNotFoundException
```

getDefaultSimulatorParameters

```
public SimulatorParameters getDefaultSimulatorParameters()
```

isStochasticSimulator

```
public boolean isStochasticSimulator()
```

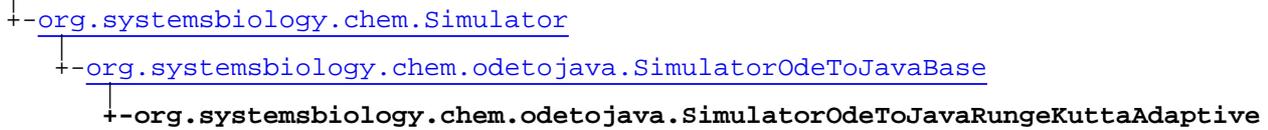
checkSimulationParametersImpl

```
public void checkSimulationParametersImpl(SimulatorParameters pSimulatorParameters,
    int pNumResultsTimePoints)
```

org.systemsbiochemistry.chem.odetojava

Class SimulatorOdeToJavaRungeKuttaAdaptive

java.lang.Object

**All Implemented interfaces:**[ISimulator](#), [IAliasableClass](#), odeToJava.modules.ODERecorder, odeToJava.modules.ODEpublic final class **SimulatorOdeToJavaRungeKuttaAdaptive**extends [SimulatorOdeToJavaBase](#)implements odeToJava.modules.ODE, odeToJava.modules.ODERecorder, [IAliasableClass](#), [ISimulator](#)

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

Constructors

SimulatorOdeToJavaRungeKuttaAdaptive

public **SimulatorOdeToJavaRungeKuttaAdaptive**()

Methods

runExternalSimulation

```
protected void runExternalSimulation(odeToJava.modules.Span pSimulationTimeSpan,
                                     double pInitialDynamicSymbolValues,
                                     double pInitialStepSize,
                                     double pMaxAllowedRelativeError,
                                     double pMaxAllowedAbsoluteError,
                                     java.lang.String pTempOutputFileName)
```

allowsInterrupt

public boolean **allowsInterrupt**()

(continued from last page)

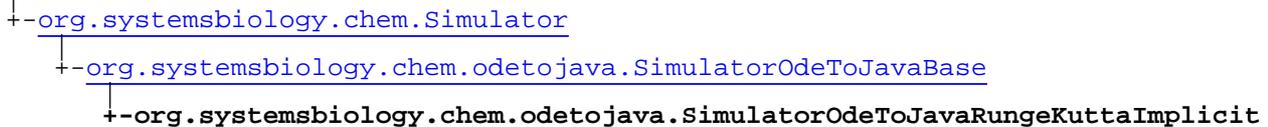
getAlias

```
public java.lang.String getAlias()
```

org.systemsbiochemistry.chem.odetojava

Class SimulatorOdeToJavaRungeKuttaImplicit

java.lang.Object



All Implemented interfaces:

[ISimulator](#), [IAliasableClass](#), odeToJava.modules.ODERecorder, odeToJava.modules.ODE

```
public final class SimulatorOdeToJavaRungeKuttaImplicit
```

```
extends SimulatorOdeToJavaBase
```

```
implements odeToJava.modules.ODE, odeToJava.modules.ODERecorder, IAliasableClass,  
ISimulator
```

Fields

CLASS_ALIAS

```
public static final java.lang.String CLASS_ALIAS
```

Constructors

SimulatorOdeToJavaRungeKuttaImplicit

```
public SimulatorOdeToJavaRungeKuttaImplicit()
```

Methods

allowsInterrupt

```
public boolean allowsInterrupt()
```

runExternalSimulation

```
protected void runExternalSimulation(odeToJava.modules.Span pSimulationTimeSpan,  
double pInitialDynamicSymbolValues,  
double pInitialStepSize,  
double pMaxAllowedRelativeError,  
double pMaxAllowedAbsoluteError,  
java.lang.String pTempOutputFileName)
```

(continued from last page)

getAlias

```
public java.lang.String getAlias()
```

Package

org.systemsbiology.chem.app

This package contains classes for a graphical user interface (GUI) application front-end for the `org.systemsbiology.chem` package.

org.systemsbiology.chem.app
Class AboutDialog

java.lang.Object
└--org.systemsbiology.chem.app.AboutDialog

public class **AboutDialog**
extends java.lang.Object

Constructors

AboutDialog

public **AboutDialog**(java.awt.Component pMainFrame)

Methods

show

public void **show**()

org.systemsbiology.chem.app

Class ChemFileFilter

java.lang.Object

└─ javax.swing.filechooser.FileFilter

└─ org.systemsbiology.chem.app.ChemFileFilter

public class **ChemFileFilter**

extends javax.swing.filechooser.FileFilter

Constructors

ChemFileFilter

public **ChemFileFilter**()

Methods

accept

public boolean **accept**(java.io.File file)

getDescription

public java.lang.String **getDescription**()

org.systemsbiology.chem.app

Class EditorPane

java.lang.Object

└-org.systemsbiology.chem.app.EditorPane

public class **EditorPane**

extends java.lang.Object

Constructors

EditorPane

public **EditorPane**(java.awt.Container pPane)

Methods

handleCut

public void **handleCut**()

handlePaste

public void **handlePaste**()

handleCopy

public void **handleCopy**()

handleResize

public void **handleResize**(int changeWidthPixels,
int changeHeightPixels)

close

public boolean **close**()

open

public void **open**()

save

```
public void save()
```

saveAs

```
public void saveAs()
```

processModel

```
public Model processModel()
```

saveEditBufferToFile

```
public void saveEditBufferToFile(java.lang.String pFileName)
```

loadFileToEditBuffer

```
public void loadFileToEditBuffer(java.lang.String pFileName)
```

org.systemsbiology.chem.app

Class MainApp

java.lang.Object

└-org.systemsbiology.chem.app.MainApp

```
public class MainApp
  extends java.lang.Object
```

Constructors

MainApp

```
public MainApp(java.lang.String pArgs)
```

Methods

getModelViewerRegistry

```
public ClassRegistry getModelViewerRegistry()
```

handleView

```
public void handleView(java.lang.String pAlias)
```

getApp

```
public static MainApp getApp()
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.chem.app

Class MainMenu

```
java.lang.Object
  |-- java.awt.Component
      |-- java.awt.Container
          |-- javax.swing.JComponent
              |-- javax.swing.JMenuBar
                  |-- org.systemsbiology.chem.app.MainMenu
```

```
public class MainMenu
  extends javax.swing.JMenuBar
```

Constructors

MainMenu

```
public MainMenu(MainApp pApp)
```

org.systemsbiology.chem.app

Class MainMenu.Menu

java.lang.Object

└--org.systemsbiology.chem.app.MainMenu.Menu

public static class **MainMenu.Menu**

extends java.lang.Object

Fields

FILE

public static final org.systemsbiology.chem.app.MainMenu.Menu **FILE**

EDIT

public static final org.systemsbiology.chem.app.MainMenu.Menu **EDIT**

TOOLS

public static final org.systemsbiology.chem.app.MainMenu.Menu **TOOLS**

HELP

public static final org.systemsbiology.chem.app.MainMenu.Menu **HELP**

Methods

toString

public java.lang.String **toString()**

get

public MainMenu.Menu **get**(java.lang.String pName)

getName

public java.lang.String **getName()**

org.systemsbiology.chem.app

Class MainMenu.MenuItem

java.lang.Object

└--org.systemsbiology.chem.app.MainMenu.MenuItem

public static class **MainMenu.MenuItem**

extends java.lang.Object

Fields

FILE_OPEN

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **FILE_OPEN**

FILE_SAVE_AS

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **FILE_SAVE_AS**

FILE_SAVE

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **FILE_SAVE**

FILE_CLOSE

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **FILE_CLOSE**

FILE_QUIT

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **FILE_QUIT**

EDIT_CUT

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **EDIT_CUT**

EDIT_COPY

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **EDIT_COPY**

EDIT_PASTE

public static final org.systemsbiology.chem.app.MainMenu.MenuItem **EDIT_PASTE**

(continued from last page)

TOOLS_EXPORT

```
public static final org.systemsbiology.chem.app.MainMenu.MenuItem TOOLS_EXPORT
```

TOOLS_SIMULATE

```
public static final org.systemsbiology.chem.app.MainMenu.MenuItem TOOLS_SIMULATE
```

TOOLS_RELOAD

```
public static final org.systemsbiology.chem.app.MainMenu.MenuItem TOOLS_RELOAD
```

TOOLS_VIEW

```
public static final org.systemsbiology.chem.app.MainMenu.MenuItem TOOLS_VIEW
```

HELP_ABOUT

```
public static final org.systemsbiology.chem.app.MainMenu.MenuItem HELP_ABOUT
```

HELP_BROWSER

```
public static final org.systemsbiology.chem.app.MainMenu.MenuItem HELP_BROWSER
```

Methods

toString

```
public java.lang.String toString()
```

get

```
public static MainMenu.MenuItem get(java.lang.String pName)
```

getName

```
public java.lang.String getName()
```

setSubMenu

```
public void setSubMenu(MainMenu.MenuItem pSubMenu,  
                       java.awt.event.ActionListener pListener)
```

getSubMenu

```
public MainMenu.MenuItem[] getSubMenu()
```

getListener

```
public java.awt.event.ActionListener getListener()
```

org.systemsbiology.chem.app

Class ModelExporter

java.lang.Object

└-org.systemsbiology.chem.app.ModelExporter

public class **ModelExporter**

extends java.lang.Object

Constructors

ModelExporter

public **ModelExporter**(java.awt.Component pMainFrame)

Methods

exportModel

public void **exportModel**(java.lang.String pAlias,
 Model pModel,
 ClassRegistry pModelExporterRegistry)
throws DataNotFoundException

org.systemsbiology.chem.app

Class ModelViewerCytoscape

java.lang.Object

└-org.systemsbiology.chem.app.ModelViewerCytoscape

All Implemented interfaces:

[IAliasableClass](#), [IModelViewer](#)

public class **ModelViewerCytoscape**

extends java.lang.Object

implements [IModelViewer](#), [IAliasableClass](#)

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

Constructors

ModelViewerCytoscape

public **ModelViewerCytoscape**()

Methods

viewModel

```
public void viewModel(Model pModel,  
                      java.lang.String pAppName)  
    throws ModelViewerException
```

org.systemsbiology.chem.app

Class ModelViewerHumanReadable

java.lang.Object

└-org.systemsbiology.chem.app.ModelViewerHumanReadable

All Implemented interfaces:

[IAliasableClass](#), [IModelViewer](#)

public class **ModelViewerHumanReadable**

extends java.lang.Object

implements [IModelViewer](#), [IAliasableClass](#)

Fields

CLASS_ALIAS

public static final java.lang.String **CLASS_ALIAS**

WIDTH

public static final int **WIDTH**

HEIGHT

public static final int **HEIGHT**

Constructors

ModelViewerHumanReadable

public **ModelViewerHumanReadable**()

Methods

viewModel

```
public void viewModel(Model pModel,  
                      java.lang.String pAppName)  
    throws ModelViewerException
```

org.systemsbiology.chem.app
Class ParserPicker

```
java.lang.Object
  |
  +--org.systemsbiology.chem.app.ParserPicker
```

```
public class ParserPicker
  extends java.lang.Object
```

Constructors

ParserPicker

```
public ParserPicker(java.awt.Component pMainFrame)
```

Methods

processFileName

```
public static java.lang.String processFileName(java.lang.String pFileName)
```

selectParserAliasManually

```
public java.lang.String selectParserAliasManually()
```

selectParserAliasFromFileName

```
public java.lang.String selectParserAliasFromFileName(java.lang.String pFileName)
```

org.systemsbiology.chem.app

Class SimulationLauncher

java.lang.Object

└--org.systemsbiology.chem.app.SimulationLauncher

public class **SimulationLauncher**

extends java.lang.Object

Constructors

SimulationLauncher

```
public SimulationLauncher(java.lang.String pAppName,  
                           Model pModel,  
                           boolean pHandleOutputInternally)
```

Creates a simulation launcher window in a JFrame.

Parameters:

`pAppName` - A string that is embedded in the title bar of the launcher frame. It should be kept short, to ensure that it displays nicely in the launcher frame title bar.

`pModel` - The {@link org.systemsbiology.chem.Model model} is a required parameter. The model may be changed by calling {@link #setModel(org.systemsbiology.chem.Model)}.

`pHandleOutputInternally` - Controls whether the launcher should handle the simulation output itself, or delegate that responsibility to the calling application. If this parameter is "true", the launcher will handle the output internally, and the caller will not have access to the simulation results in a structured format. If instead the parameter is "false", the caller will be able to access the simulation results in a structured format by calling {@link #getNextResults()}, and the launcher will not handle the simulation results data.

Methods

getSimulatorAliasesCopy

```
public java.util.Set getSimulatorAliasesCopy()
```

setCurrentDirectory

```
public void setCurrentDirectory(java.io.File pCurrentDirectory)
```

addListener

```
public void addListener(SimulationLauncher.Listener pListener)
```

Registers a {@link SimulationLauncher.Listener} object to receive events for this simulation launcher.

setModel

```
public SimulationLauncher.SetModelResult setModel(Model pModel)
```

Sets the underlying {@link org.systemsbiology.chem.Model} data structure to be `pModel`. The possible results are the enumerated class {@link SimulationLauncher.SetModelResult}.

toFront

```
public void toFront()
```

Brings the SimulationLauncher frame "to the front". Does not necessarily transfer focus to the SimulationLauncher (that depends on the window manager).

getNextResults

```
public SimulationResults getNextResults()  
    throws java.lang.IllegalStateException
```

Returns the next {@link org.systemsbiology.chem.SimulationResults} object in the queue. If the queue is empty, null is returned.

org.systemsbiology.chem.app

Interface **SimulationLauncher.Listener**

public interface **SimulationLauncher.Listener**

Methods

simulationLauncherClosing

public void **simulationLauncherClosing()**

simulationStarting

public void **simulationStarting()**

simulationEnding

public void **simulationEnding()**

org.systemsbiology.chem.app

Class **SimulationLauncher.SetModelResult**

java.lang.Object

└-org.systemsbiology.chem.app.SimulationLauncher.SetModelResult

public static class **SimulationLauncher.SetModelResult**

extends java.lang.Object

Enumerates the possible results of calling { @link #setModel(org.systemsbiology.chem.Model)}.

Fields

FAILED_CLOSED

public static final org.systemsbiology.chem.app.SimulationLauncher.SetModelResult
FAILED_CLOSED

SUCCESS

public static final org.systemsbiology.chem.app.SimulationLauncher.SetModelResult
SUCCESS

FAILED_RUNNING

public static final org.systemsbiology.chem.app.SimulationLauncher.SetModelResult
FAILED_RUNNING

org.systemsbiology.chem.app

Class SimulationLauncherCommandLine

java.lang.Object



```
public class SimulationLauncherCommandLine
```

```
extends CommandLineApp
```

Command-line interface for running a simulation.

See Also:

[org.systemsbiology.chem.Model](#), [SimulationLauncher](#)

Constructors

SimulationLauncherCommandLine

```
public SimulationLauncherCommandLine()
```

Methods

getSimulatorAliasesList

```
protected java.lang.String getSimulatorAliasesList()
```

getTimeSeriesOutputFormatAliasesList

```
protected java.lang.String getTimeSeriesOutputFormatAliasesList()
```

getParserAliasesList

```
protected java.lang.String getParserAliasesList()
```

printUsage

```
protected void printUsage(java.io.OutputStream pOutputStream)
```

handleCommandLine

```
protected void handleCommandLine(java.lang.String pArgs)
```

(continued from last page)

main

```
public static void main(java.lang.String pArgs)
```

org.systemsbiology.chem.app

Class SimulationResultsPlot

```
java.lang.Object
  |-- java.awt.Component
      |-- java.awt.Container
          |-- java.awt.Window
              |-- java.awt.Frame
                  |-- javax.swing.JFrame
                      |-- org.systemsbiology.chem.app.SimulationResultsPlot
```

public class **SimulationResultsPlot**

extends javax.swing.JFrame

Displays a plot of simulation results.

Fields

MAX_NUM_SYMBOLS_TO_PLOT

```
public static final int MAX_NUM_SYMBOLS_TO_PLOT
```

Constructors

SimulationResultsPlot

```
public SimulationResultsPlot(SimulationResults pSimulationResults,
                             java.lang.String pAppName,
                             java.lang.String pLabel)
```

Methods

handleResize

```
public void handleResize()
```

generateChart

```
protected static com.jrefinery.chart.JFreeChart generateChart(SimulationResults
pSimulationResults)
```

org.systemsbiology.chem.app

Class **SimulationResultsTable**

```
java.lang.Object
  |-- java.awt.Component
      |-- java.awt.Container
          |-- java.awt.Window
              |-- java.awt.Frame
                  |-- javax.swing.JFrame
                      |-- org.systemsbiology.chem.app.SimulationResultsTable
```

public class **SimulationResultsTable**

extends javax.swing.JFrame

Displays simulation results in a JTable

Constructors

SimulationResultsTable

```
public SimulationResultsTable(SimulationResults pSimulationResults,
                               java.lang.String pAppName,
                               java.lang.String pLabel,
                               java.text.NumberFormat pNumberFormat)
```

Package
org.systemsbiology.data

p>

org.systemsbiology.data

Class AbstractComparator

java.lang.Object

└-org.systemsbiology.data.AbstractComparator

public abstract class **AbstractComparator**
extends java.lang.Object

Constructors

AbstractComparator

public **AbstractComparator**()

Methods

compare

public abstract int **compare**(java.lang.Object p1,
java.lang.Object p2)

org.systemsbiology.data

Class DataFileDelimiter

java.lang.Object

└--org.systemsbiology.data.DataFileDelimiter

All Implemented interfaces:

java.lang.Comparable

public class **DataFileDelimiter**
extends java.lang.Object
implements java.lang.Comparable

Fields

TAB

public static final org.systemsbiology.data.DataFileDelimiter **TAB**

COMMA

public static final org.systemsbiology.data.DataFileDelimiter **COMMA**

SPACE

public static final org.systemsbiology.data.DataFileDelimiter **SPACE**

Methods

getSingle

public boolean **getSingle**()

Returns true if this delimiter type is just a single character per column. Returns false if multiple characters of this delimiter can be used to span between two adjacent columns (as with the "space" delimiter).

compareTo

public int **compareTo**(java.lang.Object pObject)

toString

public java.lang.String **toString**()

(continued from last page)

getDelimiter

```
public java.lang.String getDelimiter()
```

getName

```
public java.lang.String getName()
```

get

```
public static DataFileDelimiter get(java.lang.String pName)
```

getAll

```
public static DataFileDelimiter[] getAll()
```

getFilterRegex

```
public java.lang.String getFilterRegex()
```

getDefaultExtension

```
public java.lang.String getDefaultExtension()
```

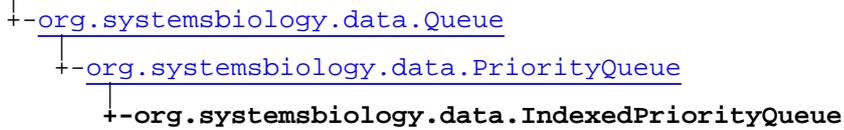
scrubIdentifier

```
public java.lang.String scrubIdentifier(java.lang.String pName)
```

org.systemsbiology.data

Class IndexedPriorityQueue

java.lang.Object



```
public class IndexedPriorityQueue
extends PriorityQueue
```

Constructors

IndexedPriorityQueue

```
public IndexedPriorityQueue(AbstractComparator pAbstractComparator)
```

IndexedPriorityQueue

```
public IndexedPriorityQueue(int pInitialCapacity,
                           AbstractComparator pAbstractComparator)
```

Methods

update

```
public void update(int pIndex,
                  java.lang.Object pValue)
    throws DataNotFoundException
```

clear

```
public void clear()
```

getNext

```
public java.lang.Object getNext()
```

peekIndex

```
public int peekIndex()
```

(continued from last page)

get

```
public java.lang.Object get(int pIndex)
```

add

```
public boolean add(java.lang.Object pElement)
```

org.systemsbiology.data
Class ListQueue

```
java.lang.Object
  |
  +--org.systemsbiology.data.Queue
      |
      +--org.systemsbiology.data.ListQueue
```

```
public final class ListQueue
extends Queue
```

Constructors

ListQueue

```
public ListQueue()
```

Methods

add

```
public boolean add(java.lang.Object pElement)
```

peekNext

```
public java.lang.Object peekNext()
```

getNext

```
public java.lang.Object getNext()
```

org.systemsbiology.data

Class MatrixString

java.lang.Object

└--org.systemsbiology.data.MatrixString

public class **MatrixString**

extends java.lang.Object

Represents a 2-dimensional matrix of String objects

Constructors

MatrixString

public **MatrixString**()

Methods

addRow

public void **addRow**(java.util.ArrayList pColumnValues)

clear

public void **clear**()

getRow

public java.lang.String[] **getRow**(int pRow)

getColumn

public java.lang.String[] **getColumn**(int pColumn)

getValueAt

public java.lang.String **getValueAt**(int pRow,
int pColumn)

getRowCount

public int **getRowCount**()

getColumnCount

```
public int getColumnCount()
```

toString

```
public java.lang.String toString()
```

toString

```
public java.lang.String toString(java.lang.String pDelimiter)
```

buildFromLineBasedStringDelimitedInput

```
public void buildFromLineBasedStringDelimitedInput(java.io.BufferedReader  
pInputReader,  
                                                    DataFileDelimiter pDelimiter)  
throws java.io.IOException,  
       InvalidInputException
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.data

Class PriorityQueue

```
java.lang.Object
  |
  +--org.systemsbiology.data.Queue
       |
       +--org.systemsbiology.data.PriorityQueue
```

Direct Known Subclasses:

[IndexedPriorityQueue](#)

```
public class PriorityQueue
extends Queue
```

Fields

mAbstractComparator

```
protected final org.systemsbiology.data.AbstractComparator mAbstractComparator
```

mRoot

```
protected org.systemsbiology.data.PriorityQueue.Node mRoot
```

Constructors

PriorityQueue

```
public PriorityQueue(AbstractComparator pAbstractComparator)
```

Methods

peekNext

```
public java.lang.Object peekNext()
```

checkIntegrity

```
public void checkIntegrity(PriorityQueue.Node pNode)
```

remove

```
protected final void remove(PriorityQueue.Node pNode,
                             AbstractComparator pAbstractComparator)
```

(continued from last page)

getNext

```
public java.lang.Object getNext()
```

insert

```
protected final static void insert(PriorityQueue.Node pTree,  
                                     PriorityQueue.Node pNode,  
                                     AbstractComparator pAbstractComparator)
```

insertRoot

```
protected final void insertRoot(PriorityQueue.Node pNode)
```

add

```
public boolean add(java.lang.Object pElement)
```

size

```
public int size()
```

toString

```
public java.lang.String toString()
```

clear

```
public void clear()
```

org.systemsbiology.data

Class PriorityQueue.Node

java.lang.Object

└-org.systemsbiology.data.PriorityQueue.Node

protected class **PriorityQueue.Node**

extends java.lang.Object

Fields

mSubtreePopulation

protected int **mSubtreePopulation**

mFirstChild

protected org.systemsbiology.data.PriorityQueue.Node **mFirstChild**

mSecondChild

protected org.systemsbiology.data.PriorityQueue.Node **mSecondChild**

mParent

protected org.systemsbiology.data.PriorityQueue.Node **mParent**

mPayload

protected java.lang.Object **mPayload**

Constructors

PriorityQueue.Node

public **PriorityQueue.Node**(java.lang.Object pPayload)

Methods

clearTreeLinks

public void **clearTreeLinks**()

(continued from last page)

org.systemsbiology.data

Class Queue

java.lang.Object

└-org.systemsbiology.data.Queue

Direct Known Subclasses:

[PriorityQueue](#), [ListQueue](#)

public abstract class **Queue**
extends java.lang.Object

Queue interface.

Constructors

Queue

public **Queue**()

Methods

add

public abstract boolean **add**(java.lang.Object pElement)

peekNext

public abstract java.lang.Object **peekNext**()

getNext

public abstract java.lang.Object **getNext**()

org.systemsbiology.data

Class SlidingWindowTimeSeriesQueue

java.lang.Object

└-org.systemsbiology.data.SlidingWindowTimeSeriesQueue

```
public class SlidingWindowTimeSeriesQueue
extends java.lang.Object
```

Implements a queue of ordered pairs of floating-point values. The first element of the ordered pair is the time, and the second element of the ordered pair is the value of some variable at that time. When the queue fills up, it start overwriting itself, discarding the oldest point first. Therefore it is a FIFO (first-in, first-out) queue. The temporal ordering of the timestamps is not enforced.

Constructors

SlidingWindowTimeSeriesQueue

```
public slidingWindowTimeSeriesQueue(int pNumTimePoints)
```

Methods

initialize

```
public void initialize(int pNumTimePoints)
```

getValue

```
public double getValue(int pIndex)
```

clear

```
public void clear()
```

hasNonzeroValue

```
public boolean hasNonzeroValue()
```

getTimeLastNonzeroValue

```
public double getTimeLastNonzeroValue()
throws java.lang.IllegalStateException
```

(continued from last page)

getLastTimePoint

```
public double getLastTimePoint()
```

getNumStoredPoints

```
public int getNumStoredPoints()
```

getTimePoint

```
public double getTimePoint(int pIndex)
```

getAverageValue

```
public double getAverageValue()
```

insertPoint

```
public void insertPoint(double pTime,  
                        double pValue)
```

getMinTime

```
public double getMinTime()
```

getTimePoints

```
public double[] getTimePoints()
```

getValues

```
public double[] getValues()
```

Package
org.systemsbiology.gui

p>

org.systemsbiology.gui

Class **ComponentUtils**

java.lang.Object

└-org.systemsbiology.gui.ComponentUtils

public class **ComponentUtils**

extends java.lang.Object

Constructors

ComponentUtils

public **ComponentUtils**()

Methods

disableDoubleClick

public static void **disableDoubleClick**(java.awt.Component c)

org.systemsbiology.gui

Class DataColumnSelector

```

java.lang.Object
  |-- java.awt.Component
      |-- java.awt.Container
          |-- java.awt.Window
              |-- java.awt.Frame
                  |-- javax.swing.JFrame
                      |-- org.systemsbiology.gui.DataColumnSelector

```

```

public class DataColumnSelector
extends javax.swing.JFrame

```

Displays a data file in tabular format, and allows the user to select columns of data to save to a new file.

Constructors

DataColumnSelector

```

public DataColumnSelector(java.lang.String pTitle,
                          MatrixString pMatrixString)

```

DataColumnSelector

```

public DataColumnSelector(java.lang.String pTitle,
                          MatrixString pMatrixString,
                          boolean pShowSaveButton)

```

Methods

setDelimiter

```

public void setDelimiter(DataFileDelimiter pDelimiter)

```

constructDataSelector

```

public static DataColumnSelector constructDataSelector(java.io.File pFile,
                                                       DataFileDelimiter pDelimiter)
                                                       throws java.io.IOException,
                                                       InvalidInputException

```

main

```

public final static void main(java.lang.String pArgs)

```

org.systemsbiology.gui

Class EmptyTableModel

```
java.lang.Object
  |
  +- javax.swing.table.AbstractTableModel
    |
    +- org.systemsbiology.gui.EmptyTableModel
```

```
public class EmptyTableModel
extends javax.swing.table.AbstractTableModel
```

Constructors

EmptyTableModel

```
public EmptyTableModel()
```

Methods

getColumnCount

```
public int getColumnCount()
```

getRowCount

```
public int getRowCount()
```

getValueAt

```
public java.lang.Object getValueAt(int pRow,
                                     int pColumn)
```

org.systemsbiology.gui

Class ExceptionNotificationOptionPane

```
java.lang.Object
  |-- java.awt.Component
      |-- java.awt.Container
          |-- javax.swing.JComponent
              |-- javax.swing.JOptionPane
                  |-- org.systemsbiology.gui.ExceptionNotificationOptionPane
```

```
public class ExceptionNotificationOptionPane
extends javax.swing.JOptionPane
```

Constructors

ExceptionNotificationOptionPane

```
public ExceptionNotificationOptionPane(java.lang.Throwable pException,
                                       java.lang.String pSummaryText)
```

ExceptionNotificationOptionPane

```
public ExceptionNotificationOptionPane(java.lang.Throwable pException)
```

Methods

handleDetailedButton

```
public void handleDetailedButton()
```

org.systemsbiology.gui

Class ExHelpSearchNavigator

```
java.lang.Object
  |
  +- javax.help.NavigatorView
      |
      +- javax.help.SearchView
          |
          +- org.systemsbiology.gui.ExHelpSearchNavigator
```

```
public class ExHelpSearchNavigator
extends javax.help.SearchView
```

Constructors

ExHelpSearchNavigator

```
public ExHelpSearchNavigator(javax.help.HelpSet hs,
                             java.lang.String name,
                             java.lang.String label,
                             java.util.Hashtable params)
```

ExHelpSearchNavigator

```
public ExHelpSearchNavigator(javax.help.HelpSet hs,
                             java.lang.String name,
                             java.lang.String label,
                             java.util.Locale locale,
                             java.util.Hashtable params)
```

Methods

createNavigator

```
public java.awt.Component createNavigator(javax.help.HelpModel model)
```

org.systemsbiology.gui
Class FileChooser

```
java.lang.Object
  |
  +--java.awt.Component
        |
        +--java.awt.Container
              |
              +--javax.swing.JComponent
                    |
                    +--javax.swing.JFileChooser
                          |
                          +--org.systemsbiology.gui.FileChooser
```

```
public class FileChooser
  extends javax.swing.JFileChooser
```

Constructors

FileChooser

```
public FileChooser()
```

FileChooser

```
public FileChooser(java.io.File pCurrentDirectory)
```

Methods

handleOutputFileAlreadyExists

```
public static boolean handleOutputFileAlreadyExists(java.awt.Component pFrame,
                                                    java.lang.String pOutputFileName)
```

org.systemsbiology.gui
Class FramePlacer

java.lang.Object
└─org.systemsbiology.gui.FramePlacer

public class **FramePlacer**
extends java.lang.Object

Methods for selecting the default location for a GUI frame.

Constructors

FramePlacer

public **FramePlacer**()

Methods

getScreenSize

public static java.awt.Dimension **getScreenSize**()

placeInCascadeFormat

public void **placeInCascadeFormat**(javax.swing.JFrame pFrame)

Arranges subsequent frames in a diagonal cascade (down and to the right).

placeInCenterOfScreen

public static java.awt.Point **placeInCenterOfScreen**(int pFrameWidth,
int pFrameHeight)

placeInCenterOfScreen

public static void **placeInCenterOfScreen**(javax.swing.JFrame pFrame)

Places a frame in the center of the screen

org.systemsbiology.gui

Class HelpBrowser

java.lang.Object

|
+-org.systemsbiology.gui.HelpBrowser

public class **HelpBrowser**extends java.lang.Object

Constructors

HelpBrowser

```
public HelpBrowser(java.awt.Component pMainFrame,  
                  java.lang.String pHelpSetName,  
                  java.lang.String pAppName)
```

Methods

setFrameTitle

```
public void setFrameTitle(java.lang.String pTitle)
```

setFrameSize

```
public void setFrameSize(java.awt.Dimension pFrameSize)
```

setFrameLocation

```
public void setFrameLocation(java.awt.Point pLocation)
```

displayHelpBrowser

```
public void displayHelpBrowser(java.lang.String pMapID,  
                               java.lang.String pView)
```

org.systemsbiology.gui

Class **IconFactory**

java.lang.Object

└-org.systemsbiology.gui.**IconFactory**

public class **IconFactory**

extends java.lang.Object

Constructors

IconFactory

public **IconFactory**()

Methods

getIconByName

public static javax.swing.ImageIcon **getIconByName**(java.lang.String pIconName)

org.systemsbiology.gui

Class ImageTransferHandler

```
java.lang.Object
  |
  +- javax.swing.TransferHandler
    |
    +- org.systemsbiology.gui.ImageTransferHandler
```

```
public class ImageTransferHandler
  extends javax.swing.TransferHandler
```

Constructors

ImageTransferHandler

```
public ImageTransferHandler()
```

Methods

getSourceActions

```
public int getSourceActions(javax.swing.JComponent c)
```

canImport

```
public boolean canImport(javax.swing.JComponent comp,
                          java.awt.datatransfer.DataFlavor flavor)
```

checkDoesSystemClipboardSupportImageTransfer

```
public static boolean checkDoesSystemClipboardSupportImageTransfer()
```

createTransferable

```
public java.awt.datatransfer.Transferable createTransferable(javax.swing.JComponent
  comp)
```

org.systemsbiology.gui

Class MatrixStringSelectorDialog

```
java.lang.Object
  |-- java.awt.Component
      |-- java.awt.Container
          |-- java.awt.Window
              |-- java.awt.Dialog
                  |-- javax.swing.JDialog
                      |-- org.systemsbiology.gui.MatrixStringSelectorDialog
```

```
public class MatrixStringSelectorDialog
extends javax.swing.JDialog
```

Constructors

MatrixStringSelectorDialog

```
public MatrixStringSelectorDialog(java.awt.Frame pParent,
                                   java.lang.String pTitle,
                                   boolean pModal,
                                   MatrixString pMatrixString,
                                   boolean pFirstRowIsTitles)
```

Methods

setDelimiter

```
public void setDelimiter(DataFileDelimiter pDelimiter)
```

setMultipleSelectionsAllowed

```
public void setMultipleSelectionsAllowed(boolean pMultipleSelectionsAllowed)
```

handleCancel

```
public void handleCancel()
```

addApproveListener

```
public void addApproveListener(java.awt.event.ActionListener pActionListener)
```

(continued from last page)

getSelectedColumns

```
public java.lang.Boolean[] getSelectedColumns()
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.gui

Class **RegexFileFilter**

java.lang.Object

├--javax.swing.filechooser.FileFilter

└--org.systemsbiology.gui.RegexFileFilter

public class **RegexFileFilter**

extends javax.swing.filechooser.FileFilter

Constructors

RegexFileFilter

```
public RegexFileFilter(java.lang.String pRegex,  
                      java.lang.String pDescription)
```

Methods

accept

```
public boolean accept(java.io.File file)
```

getDescription

```
public java.lang.String getDescription()
```

org.systemsbiology.gui

Class SimpleTextArea

```
java.lang.Object
  |
  +-- java.awt.Component
        |
        +-- java.awt.Container
              |
              +-- javax.swing.JComponent
                    |
                    +-- javax.swing.text.JTextComponent
                          |
                          +-- javax.swing.JTextArea
                                |
                                +-- org.systemsbiology.gui.SimpleTextArea
```

```
public class SimpleTextArea
extends javax.swing.JTextArea
```

Constructors

SimpleTextArea

```
public SimpleTextArea(java.lang.String pText)
```

org.systemsbiology.gui

Class SortStatus

java.lang.Object

└-org.systemsbiology.gui.SortStatus

public class **SortStatus**
extends java.lang.Object

Fields

mCode

protected int **mCode**

CODE_NONE

public static final int **CODE_NONE**

CODE_ASCENDING

public static final int **CODE_ASCENDING**

CODE_DESCENDING

public static final int **CODE_DESCENDING**

NONE

public static final org.systemsbiology.gui.SortStatus **NONE**

ASCENDING

public static final org.systemsbiology.gui.SortStatus **ASCENDING**

DESCENDING

public static final org.systemsbiology.gui.SortStatus **DESCENDING**

Methods

(continued from last page)

getCode

```
public int getCode()
```

getName

```
public java.lang.String getName()
```

getNextSortStatusInCycle

```
public SortStatus getNextSortStatusInCycle()
```

getAll

```
public static SortStatus\[\] getAll()
```

Package

org.systemsbiology.inference

/p>

org.systemsbiology.inference

Class DataManagerDriver

```
java.lang.Object
  |
  +--org.systemsbiology.inference.DataManagerDriver
```

```
public class DataManagerDriver
  extends java.lang.Object
```

A graphical tool for merging {[@link ObservationsData](#)} data files. Rows are specific elements, and columns are specific evidence types. The entire data table may be saved to a file, or a set of columns may be selected and saved to a file.

Constructors

DataManagerDriver

```
public DataManagerDriver()
```

Methods

initialize

```
public void initialize(java.awt.Container pContentPane,
                       java.awt.Component pParent,
                       java.lang.String pProgramName)
```

savePreferences

```
public void savePreferences()
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.inference

Class DataNormalizationMethod

java.lang.Object

└-org.systemsbiology.inference.DataNormalizationMethod

public class **DataNormalizationMethod**

extends java.lang.Object

Species how the { @link DataNormalizer } should normalize a matrix of raw observations.

Fields

QUANTILE

public static final org.systemsbiology.inference.DataNormalizationMethod **QUANTILE**

Methods

get

public static [DataNormalizationMethod](#) **get**(java.lang.String pName)

getName

public java.lang.String **getName**()

getAll

public static [DataNormalizationMethod\[\]](#) **getAll**()

org.systemsbiology.inference

Class DataNormalizationScale

java.lang.Object

└--org.systemsbiology.inference.DataNormalizationScale

public class **DataNormalizationScale**

extends java.lang.Object

Specifies the scale to be used for the {@link DataNormalizer}. This is essentially an enumeration class. An object of this class is a field in the {@link DataNormalizerParams} class, which is passed to the quantile normalizer. Note that if you choose the `DataNormalizationScale.LOGARITHM` object, you should set the `mFixNonpositiveValues` field in the {@link DataNormalizerParams} object, to ensure that any observations less than or equal to zero are fixed (by a global additive shift of the raw observation values) prior to the logarithmic rescaling.

Fields

LOGARITHM

public static final org.systemsbiology.inference.DataNormalizationScale **LOGARITHM**

NORM_ONLY

public static final org.systemsbiology.inference.DataNormalizationScale **NORM_ONLY**

Methods

get

public static [DataNormalizationScale](#) **get**(java.lang.String pName)

getName

public java.lang.String **getName**()

getAll

public static [DataNormalizationScale\[\]](#) **getAll**()

allowsNonpositiveArgument

public boolean **allowsNonpositiveArgument**()

org.systemsbiology.inference

Class DataNormalizer

java.lang.Object

```

  |
  +--org.systemsbiology.inference.DataNormalizer

```

public class **DataNormalizer**

extends java.lang.Object

Performs a normalization of a matrix of raw observations, so that each column of the matrix is consistent, based on the {@link DataNormalizationMethod} passed in the {@link DataNormalizerParams} object. One (and currently, the only) method is {@link DataNormalizationMethod#QUANTILE}, which implements a quantile normalization such that the medians of all the columns will be the same. Here, normalization means assigning a "normalized" observation value to each "raw" observation value in the matrix. Consolidation of the data (e.g., averaging) is a separate procedure. The normalization is performed after a rescaling of the raw observations in accordance of the scale parameter {@link DataNormalizationScale}. All parameters for the normalization are contained in the {@link DataNormalizerParams} class. Results are stored in the {@link DataNormalizerResults} object, which must have a pre-allocated `cern.colt.matrix.ObjectMatrix2D` object `mNormalizedObservations` to hold the normalized observation values. For the {@link DataNormalizationMethod#QUANTILE} method, the quantile normalization algorithm used is based on a prototype written by Daehee Hwang at Institute for Systems Biology, and it is similar to the quantile normalization algorithm proposed by Bolstad et al. in their paper Bolstad, B.M., Irizarry R. A., Astrand M., and Speed, T.P. (2003), "A Comparison of Normalization Methods for High Density Oligonucleotide Array Data Based on Bias and Variance." *Bioinformatics* **19**(2):185-193 Note that only the quantile normalization step of the RMA (Robust Multi-Chi Average) procedure is implemented in this class; background adjustment is not implemented here, and is assumed to have been applied to the raw observations before this class's normalization function is to be executed.

Constructors

DataNormalizer

```
public DataNormalizer()
```

Methods

rescaleRawObservations

```
public void rescaleRawObservations(DataNormalizationScale pScale)
```

normalize

```
public void normalize(cern.colt.matrix.ObjectMatrix2D pRawObservations,
                    DataNormalizerParams pParams,
                    DataNormalizerResults pResults)
```

normalize

```
public DataNormalizerResults normalize(cern.colt.matrix.ObjectMatrix2D
pRawObservations,
                                         DataNormalizerParams pParams)
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.inference

Class DataNormalizerDriver

java.lang.Object

└--org.systemsbiology.inference.DataNormalizerDriver

public class **DataNormalizerDriver**

extends java.lang.Object

A graphical user interface for normalizing raw observations using the {@link DataNormalizer}. The data may be optionally rescaled before the normalization, using the {@link DataNormalizationScale} parameter to the {@link DataNormalizerParams} object. The algorithms used in this class are based on the ideas and designs of Daehee Hwang at Institute for Systems Biology.

Constructors

DataNormalizerDriver

```
public DataNormalizerDriver()
```

Methods

initialize

```
public void initialize(java.awt.Container pContentPane,  
                      java.awt.Component pParent,  
                      java.lang.String pProgramName)
```

savePreferences

```
public void savePreferences()
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.inference

Class DataNormalizerParams

java.lang.Object

└--org.systemsbiology.inference.DataNormalizerParams

public class **DataNormalizerParams**

extends java.lang.Object

Describes how the {@link DataNormalizer} is to normalize the data. In particular, the {@link DataNormalizationScale} is defined, as well as the error tolerance (only applicable if there is missing data), and where non-positive values should be fixed by a uniform additive adjustment to the data prior to the rescaling.

Fields

mMethod

public org.systemsbiology.inference.DataNormalizationMethod **mMethod**

mScale

public org.systemsbiology.inference.DataNormalizationScale **mScale**

mErrorTolerance

public java.lang.Double **mErrorTolerance**

mFixNonpositiveValues

public boolean **mFixNonpositiveValues**

mMaxIterations

public java.lang.Integer **mMaxIterations**

Constructors

DataNormalizerParams

public **DataNormalizerParams**()

org.systemsbiology.inference

Class DataNormalizerResults

java.lang.Object

└─org.systemsbiology.inference.DataNormalizerResults

public class **DataNormalizerResults**

extends java.lang.Object

The results of a call to the {@link DataNormalizer}. Contains the normalized observations, which are stored in a matrix `cern.colt.matrix.ObjectMatrix2D`, where null entries represent missing data (i.e., observations that were missing in the un-normalized raw observations that were originally passed to the quantile normalizer). The number of iterations needed for the quantile normalizer to converge is returned in the field `mNumIterations` (if there is no missing data, this field will always be unity). The final error field `mFinalError` is set to the value of the final fractional error in the normalization (coming from estimation of missing data); this field is set to null if there is no missing data.

Fields

mNormalizedObservations

public `cern.colt.matrix.ObjectMatrix2D` **mNormalizedObservations**

mNumIterations

public int **mNumIterations**

mFinalError

public `java.lang.Double` **mFinalError**

Constructors

DataNormalizerResults

public **DataNormalizerResults**()

org.systemsbiology.inference

Class EvidenceWeightedInferer

java.lang.Object

└-org.systemsbiology.inference.EvidenceWeightedInferer

```
public final class EvidenceWeightedInferer
extends java.lang.Object
```

An implementation of the Pointillist algorithm for inferring the set elements affected by a perturbation of a biological system, based on multiple types of evidence. The algorithm computes the combined probability that an element would have a given set of significance values, by chance, if it were not in the set of true affected elements. This algorithm was designed by Daehee Hwang at Institute for Systems Biology.

Fields

MIN_NUM_BINS

```
public static final int MIN_NUM_BINS
```

Constructors

EvidenceWeightedInferer

```
public EvidenceWeightedInferer()
```

Methods

findAffectedElements

```
public void findAffectedElements(cern.colt.matrix.DoubleMatrix2D pSignificances,
                                int pNumBins,
                                double pInitialCutoff,
                                double pCombinedSignificanceQuantileCutoff,
                                double pFractionToRemove,
                                double pMinFractionalCostChange,
                                double pSmoothingLength,
                                int pMaxIterations,
                                EvidenceWeightType pWeightType,
                                EvidenceWeightedInfererResults pRetResults)
    throws AccuracyException
```

Parameters:

```
pSignificances
pNumBins
pInitialCutoff
pCombinedSignificanceQuantileCutoff
pFractionToRemove
pMinFractionalCostChange
pSmoothingLength
```

(continued from last page)

pMaxIterations - To have no maximum number of iterations, pass 0 for this parameter. This parameter may not be negative.

pWeightType

pRetResults

Exceptions:

AccuracyException

findAffectedElements

```
public void findAffectedElements(cern.colt.matrix.DoubleMatrix2D pSignificances,  
                                EvidenceWeightedInfererParams pParams,  
                                EvidenceWeightedInfererResults pResults)  
    throws AccuracyException
```

findAffectedElements

```
public EvidenceWeightedInfererResults findAffectedElements  
(cern.colt.matrix.DoubleMatrix2D pSignificances,  
 EvidenceWeightedInfererParams pParams)  
    throws AccuracyException
```

org.systemsbiology.inference

Class EvidenceWeightedInfererDriver

java.lang.Object

└-org.systemsbiology.inference.EvidenceWeightedInfererDriver

```
public class EvidenceWeightedInfererDriver
extends java.lang.Object
```

Graphical user interface for the Pointillist algorithm for inferring the set of elements affected by a perturbation of a biological system. The algorithm computes the combined significance, which is analogous (but not identical) to the joint probability that an element would have a given set of significance values, by chance, if it were not in the set of true affected elements. This algorithm was designed by Daehee Hwang at Institute for Systems Biology.

Fields

ALPHA_VALUE_DEVIATION_WARNING_THRESHOLD

```
public static final double ALPHA_VALUE_DEVIATION_WARNING_THRESHOLD
```

DEFAULT_NUM_BINS

```
public static final int DEFAULT_NUM_BINS
```

DEFAULT_INITIAL_SIGNIFICANCE_CUTOFF

```
public static final double DEFAULT_INITIAL_SIGNIFICANCE_CUTOFF
```

DEFAULT_QUANTILE_THRESHOLD

```
public static final double DEFAULT_QUANTILE_THRESHOLD
```

DEFAULT_SEPARATION_THRESHOLD

```
public static final double DEFAULT_SEPARATION_THRESHOLD
```

DEFAULT_EVIDENCE_WEIGHT_TYPE

```
public static final org.systemsbiology.inference.EvidenceWeightType
DEFAULT_EVIDENCE_WEIGHT_TYPE
```

DEFAULT_DATA_FILE_DELIMITER

```
public static final org.systemsbiology.data.DataFileDelimiter
DEFAULT_DATA_FILE_DELIMITER
```

(continued from last page)

Constructors

EvidenceWeightedInfererDriver

```
public EvidenceWeightedInfererDriver()
```

Methods

initialize

```
public void initialize(java.awt.Container pContentPane,  
                       java.awt.Component pParent,  
                       java.lang.String pProgramName)
```

savePreferences

```
public void savePreferences()
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.inference

Class EvidenceWeightedInfererParams

java.lang.Object

└--org.systemsbiology.inference.EvidenceWeightedInfererParams

public class **EvidenceWeightedInfererParams**

extends java.lang.Object

Data structure that holds the parameters that are passed to the {@link EvidenceWeightedInferer}, to describe how the Pointillist algorithm should run. The parameters include the number of bins used to calculate the nonparametric distribution, the initial cutoff for the evidence-specific significances, the final cutoff for the combined significances, the smoothing length for the log-ignificance distribution, and the type of formula to be used to compute the evidence-specific weights.

Constructors

EvidenceWeightedInfererParams

public **EvidenceWeightedInfererParams**()

Methods

setMaxIterations

public void **setMaxIterations**(java.lang.Integer pMaxIterations)

getMaxIterations

public java.lang.Integer **getMaxIterations**()

setNumBins

public void **setNumBins**(java.lang.Integer pNumBins)

getNumBins

public java.lang.Integer **getNumBins**()

setInitialSignificanceCutoff

public void **setInitialSignificanceCutoff**(java.lang.Double pInitialSignificanceCutoff)

(continued from last page)

getInitialSignificanceCutoff

```
public java.lang.Double getInitialSignificanceCutoff()
```

setCombinedSignificanceCutoff

```
public void setCombinedSignificanceCutoff(java.lang.Double  
pCombinedSignificanceQuantileCutoff)
```

getCombinedSignificanceCutoff

```
public java.lang.Double getCombinedSignificanceCutoff()
```

setFractionToRemove

```
public void setFractionToRemove(java.lang.Double pFractionToRemove)
```

getFractionToRemove

```
public java.lang.Double getFractionToRemove()
```

setMinFractionalCostChange

```
public void setMinFractionalCostChange(java.lang.Double pMinFractionalCostChange)
```

getMinFractionalCostChange

```
public java.lang.Double getMinFractionalCostChange()
```

setSmoothingLength

```
public void setSmoothingLength(java.lang.Double pSmoothingLength)
```

getSmoothingLength

```
public java.lang.Double getSmoothingLength()
```

setEvidenceWeightType

```
public void setEvidenceWeightType(EvidenceWeightType pEvidenceWeightType)
```

getEvidenceWeightType

```
public EvidenceWeightType getEvidenceWeightType()
```

(continued from last page)

org.systemsbiology.inference

Class EvidenceWeightedInfererResults

java.lang.Object

└--org.systemsbiology.inference.EvidenceWeightedInfererResults

public class **EvidenceWeightedInfererResults**

extends java.lang.Object

A data structure that holds the results of a call to the {@link EvidenceWeightedInferer}. A global set of element names was passed to the inferer. The boolean values indicating whether each element is a member of the putative set of affected elements, is included in this data structure of results from the EvidenceWeightedInferer. The combined effective significances are included as well.

Fields

mSignificanceDistributionSeparation

public double **mSignificanceDistributionSeparation**

mNumIterations

public int **mNumIterations**

mAlphaParameter

public double **mAlphaParameter**

mCombinedEffectiveSignificances

public double **mCombinedEffectiveSignificances**

mAffectedElements

public boolean **mAffectedElements**

mWeights

public double **mWeights**

mNumAffected

public int **mNumAffected**

mIterationSummaryAffected

```
public int mIterationSummaryAffected
```

mIterationSummaryFalseNegatives

```
public int mIterationSummaryFalseNegatives
```

Constructors

EvidenceWeightedInfererResults

```
public EvidenceWeightedInfererResults()
```

org.systemsbiology.inference

Class EvidenceWeightType

java.lang.Object

└--org.systemsbiology.inference.EvidenceWeightType

All Implemented interfaces:

java.lang.Comparable

public class **EvidenceWeightType**

extends java.lang.Object

implements java.lang.Comparable

Enumerates the formulas that may be used for computing the evidence-specific weights. Passed to the {@link EvidenceWeightedInferer} as a parameter.

Fields

CODE_LINEAR

public static final int **CODE_LINEAR**

CODE_POWER

public static final int **CODE_POWER**

CODE_UNIFORM

public static final int **CODE_UNIFORM**

LINEAR

public static final org.systemsbiology.inference.EvidenceWeightType **LINEAR**

POWER

public static final org.systemsbiology.inference.EvidenceWeightType **POWER**

UNIFORM

public static final org.systemsbiology.inference.EvidenceWeightType **UNIFORM**

Methods

(continued from last page)

getCode

```
public int getCode()
```

toString

```
public java.lang.String toString()
```

get

```
public static EvidenceWeightType get(java.lang.String pName)
```

compareTo

```
public int compareTo(java.lang.Object pObject)
```

getName

```
public java.lang.String getName()
```

getAll

```
public static EvidenceWeightType\[\] getAll()
```

org.systemsbiology.inference

Class ObservationsData

java.lang.Object

└--org.systemsbiology.inference.ObservationsData

All Implemented interfaces:

java.lang.Cloneable

```
public class ObservationsData
  extends java.lang.Object
  implements java.lang.Cloneable
```

A data structure containing a set of observations for M elements, and N types of evidence. Missing observations are stored as a null in the data matrix, and can be denoted by either an empty cell or the string "null". Non-missing observations are stored as a {@link Double} object. This class is used by the {@link SignificanceCalculatorDriver} class.

Fields

NULL_OBSERVATION_STRING

```
public static final java.lang.String NULL_OBSERVATION_STRING
```

Constructors

ObservationsData

```
public ObservationsData()
```

ObservationsData

```
public ObservationsData(cern.colt.matrix.ObjectMatrix2D pObservations,
                        java.lang.String pElementNames,
                        java.lang.String pEvidenceNames)
```

Methods

setElementName

```
public void setElementName(int pElementNumber,
                            java.lang.String pElementName)
```

setEvidenceName

```
public void setEvidenceName(int pEvidenceNumber,
                              java.lang.String pEvidenceName)
```

mergeDataArray

```
public void mergeDataArray(ObservationsData pObservationsDataArray,  
                           boolean pAllowDuplicates)
```

clone

```
public java.lang.Object clone()
```

getNumElements

```
public int getNumElements()
```

getNumEvidences

```
public int getNumEvidences()
```

getEvidenceName

```
public java.lang.String getEvidenceName(int pColumn)
```

getElementName

```
public java.lang.String getElementName(int pRow)
```

getElementNames

```
public java.lang.String[] getElementNames()
```

getMissingDataRate

```
public double getMissingDataRate()
```

getEvidenceNames

```
public java.lang.String[] getEvidenceNames()
```

setValueAt

```
public void setValueAt(int pRow,  
                       int pColumn,  
                       java.lang.Double pValue)
```

getValueAt

```
public java.lang.Double getValueAt(int pRow,  
                                     int pColumn)
```

getColumn

```
public java.lang.Double[] getColumn(int pColumn)
```

getNonMissingColumnVals

```
public cern.colt.list.DoubleArrayList getNonMissingColumnVals(int pColumn)
```

writeToFile

```
public void writeToFile(java.io.PrintWriter pPrintWriter,  
                        DataFileDelimiter pDelimiter,  
                        java.text.NumberFormat pNumberFormat)
```

loadFromFile

```
public void loadFromFile(java.io.BufferedReader pBufferedReader,  
                        DataFileDelimiter pDelimiter)  
    throws java.io.IOException,  
           InvalidInputException
```

getNumObservations

```
public int getNumObservations(int pEvidenceNum)
```

org.systemsbiology.inference

Class ObservationsTableModel

java.lang.Object

├--javax.swing.table.AbstractTableModel

└--org.systemsbiology.inference.ObservationsTableModel

```
public class ObservationsTableModel
extends javax.swing.table.AbstractTableModel
```

A graphical user interface table model for {@link ObservationsData}. Allows sorting of rows by element name, and sorting of columns by evidence name.

Fields

COLUMN_NAME_ELEMENT

```
public static final java.lang.String COLUMN_NAME_ELEMENT
```

Constructors

ObservationsTableModel

```
public ObservationsTableModel(ObservationsData pObservationsData)
```

Methods

setElementNamesEditable

```
public void setElementNamesEditable(boolean pElementNamesEditable)
```

setEvidenceNamesEditable

```
public void setEvidenceNamesEditable(boolean pEvidenceNamesEditable)
```

setCellValuesEditable

```
public void setCellValuesEditable(boolean pCellValuesEditable)
```

setObservationsData

```
public void setObservationsData(ObservationsData pObservationsData)
```

setElementSortStatus

```
public void setElementSortStatus(SortStatus pSortStatus)
```

setEvidenceSortStatus

```
public void setEvidenceSortStatus(SortStatus pSortStatus)
```

getEvidenceSortStatus

```
public SortStatus getEvidenceSortStatus()
```

getElementSortStatus

```
public SortStatus getElementSortStatus()
```

getRowCount

```
public int getRowCount()
```

getColumnCount

```
public int getColumnCount()
```

getColumnName

```
public java.lang.String getColumnName(int pColumn)
```

getValueAtNoFormatting

```
public java.lang.Object getValueAtNoFormatting(int pRow,  
                                                int pColumn)
```

isCellEditable

```
public boolean isCellEditable(int pRow,  
                               int pColumn)
```

setValueAt

```
public void setValueAt(java.lang.Object pValue,  
                       int pRow,  
                       int pColumn)
```

getValueAt

```
public java.lang.Object getValueAt(int pRow,  
                                     int pColumn)
```

org.systemsbiology.inference

Class PreferencesHandler

```
java.lang.Object
  |
  +--org.systemsbiology.inference.PreferencesHandler
```

```
public class PreferencesHandler
extends java.lang.Object
```

Gets and sets user preferences for the org.systemsbiology.inference package GUI programs.

Fields

KEY_DELIMITER

```
public static final java.lang.String KEY_DELIMITER
```

Constructors

PreferencesHandler

```
public PreferencesHandler()
```

Methods

setPreference

```
public void setPreference(java.lang.String pKey,
                          java.lang.String pValue)
```

getPreference

```
public java.lang.String getPreference(java.lang.String pKey,
                                      java.lang.String pDefault)
```

flush

```
public void flush()
    throws java.util.prefs.BackingStoreException
```

org.systemsbiology.inference

Class SignificanceCalculationMethod

java.lang.Object

└-org.systemsbiology.inference.SignificanceCalculationMethod

All Implemented interfaces:

java.lang.Comparable

public class **SignificanceCalculationMethod**

extends java.lang.Object

implements java.lang.Comparable

Enumerates the possible formulas for computing the significance of an observation, based on the distribution of "negative control" observations. Passed to the {@link SignificanceCalculator} as a parameter.

Fields

CODE_CDF_NONPARAMETRIC

public static final int **CODE_CDF_NONPARAMETRIC**

CODE_PDF_NONPARAMETRIC

public static final int **CODE_PDF_NONPARAMETRIC**

CODE_CDF_PARAMETRIC

public static final int **CODE_CDF_PARAMETRIC**

CODE_PDF_PARAMETRIC

public static final int **CODE_PDF_PARAMETRIC**

CDF_NONPARAMETRIC

public static final org.systemsbiology.inference.SignificanceCalculationMethod
CDF_NONPARAMETRIC

PDF_NONPARAMETRIC

public static final org.systemsbiology.inference.SignificanceCalculationMethod
PDF_NONPARAMETRIC

(continued from last page)

PDF_PARAMETRIC

```
public static final org.systemsbiology.inference.SignificanceCalculationMethod  
PDF_PARAMETRIC
```

CDF_PARAMETRIC

```
public static final org.systemsbiology.inference.SignificanceCalculationMethod  
CDF_PARAMETRIC
```

Methods

getCode

```
public int getCode()
```

toString

```
public java.lang.String toString()
```

compareTo

```
public int compareTo(java.lang.Object pObject)
```

get

```
public static SignificanceCalculationMethod get(java.lang.String pName)
```

getAll

```
public static SignificanceCalculationMethod\[\] getAll()
```

getName

```
public java.lang.String getName()
```

org.systemsbiology.inference

Class SignificanceCalculationResults

java.lang.Object

└-org.systemsbiology.inference.SignificanceCalculationResults

public class **SignificanceCalculationResults**

extends java.lang.Object

Data structure containing the results of a call to the {@link SignificanceCalculator}. It contains the significance values of the observations that were passed to the significance calculator.

Fields

mSignificances

public double **mSignificances**

mReducedChiSquare

public double **mReducedChiSquare**

mBestFitDistribution

public org.systemsbiology.math.probability.IContinuousDistribution
mBestFitDistribution

Constructors

SignificanceCalculationResults

public **SignificanceCalculationResults**()

SignificanceCalculationResults

public **SignificanceCalculationResults**(int pNumObservations)

org.systemsbiology.inference

Class SignificanceCalculator

java.lang.Object

└-org.systemsbiology.inference.SignificanceCalculator

```
public class SignificanceCalculator
extends java.lang.Object
```

Based on a set of "negative control" observations, computes the significance of each of a set of observations. The set of control observations may or may not be identically the same as the observations for which the significances are to be computed, which allows for computing the significances of a set of observations in the absence of separate "negative control" measurements. The method used for computing the significance depends on the {@link SignificanceCalculationMethod} passed as a parameter. The algorithms used in this class are based on the ideas and designs of Daehee Hwang at Institute for Systems Biology.

Fields

DEFAULT_MISSING_DATA_SIGNIFICANCE

```
public static final double DEFAULT_MISSING_DATA_SIGNIFICANCE
```

Constructors

SignificanceCalculator

```
public SignificanceCalculator()
```

Methods

setMissingDataSignificance

```
public void setMissingDataSignificance(double pMissingDataSignificance)
```

computeSignificanceNonParametric

```
public static double computeSignificanceNonParametric(int pNonParametricMethod,
                                                       double pMean,
                                                       double pVariance,
                                                       double x)
```

(continued from last page)

calculateSignificancesNonParametric

```
public void calculateSignificancesNonParametric(java.lang.Double pObservations,
                                                java.lang.Double pControlData,
                                                int pNumBins,
                                                boolean pSingleTailed,
                                                double pSmoothingLength,
                                                int pNonParametricMethodCode,
                                                SignificanceCalculationResults
pRetResults)
```

calculateSignificancesParametric

```
public void calculateSignificancesParametric(java.lang.Double pObservations,
                                             java.lang.Double pControlData,
                                             int pNumBins,
                                             boolean pSingleTailed,
                                             double pMaxChiSquare,
                                             int pParametricMethodCode,
                                             SignificanceCalculationResults
pRetResults)
throws AccuracyException
```

calculateSignificances

```
public void calculateSignificances(java.lang.Double pObservations,
                                   java.lang.Double pControlData,
                                   SignificanceCalculatorParams pParams,
                                   SignificanceCalculationResults pResults)
throws AccuracyException
```

calculateSignificances

```
public SignificanceCalculationResults calculateSignificances(java.lang.Double
pObservations,
                                                               java.lang.Double
pControlData,
SignificanceCalculatorParams pParams)
throws AccuracyException
```

org.systemsbiology.inference

Class SignificanceCalculatorDriver

java.lang.Object

└-org.systemsbiology.inference.SignificanceCalculatorDriver

public class **SignificanceCalculatorDriver**

extends java.lang.Object

Graphical user interface for calling the { @link SignificanceCalculator} to obtain significances of observations. Multiple types of evidence can be processed in a single run of the program, with distinct parameter values for the different types of evidence. The algorithms used in this class are based on the ideas and designs of Daehee Hwang at Institute for Systems Biology.

Constructors

SignificanceCalculatorDriver

public **SignificanceCalculatorDriver**()

Methods

initialize

```
public void initialize(java.awt.Container pContentPane,  
                       java.awt.Component pParent,  
                       java.lang.String pProgramName)
```

savePreferences

```
public void savePreferences()
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.inference

Class SignificanceCalculatorParams

java.lang.Object

└--org.systemsbiology.inference.SignificanceCalculatorParams

```
public class SignificanceCalculatorParams
extends java.lang.Object
```

The set of parameters that are passed to the {@link SignificanceCalculator}, describing how the significance calculation should be performed. The parameters include the number of bins for the nonparametric distribution, the smoothing length for the kernel density, whether the distribution is single-tailed or two-tailed, etc.

Constructors

SignificanceCalculatorParams

```
public SignificanceCalculatorParams()
```

Methods

setSignificanceCalculationMethod

```
public void setSignificanceCalculationMethod(SignificanceCalculationMethod pFormula)
```

getSignificanceCalculationMethod

```
public SignificanceCalculationMethod getSignificanceCalculationMethod()
```

setSingleTailed

```
public void setSingleTailed(java.lang.Boolean pSingleTailed)
```

getSingleTailed

```
public java.lang.Boolean getSingleTailed()
```

setNumBins

```
public void setNumBins(java.lang.Integer pNumBins)
```

setSmoothingLength

```
public void setSmoothingLength(java.lang.Double pSmoothingLength)
```

(continued from last page)

setMaxReducedChiSquare

```
public void setMaxReducedChiSquare(java.lang.Double pMaxReducedChiSquare)
```

getMaxReducedChiSquare

```
public java.lang.Double getMaxReducedChiSquare()
```

getSmoothingLength

```
public java.lang.Double getSmoothingLength()
```

getNumBins

```
public java.lang.Integer getNumBins()
```

org.systemsbiology.inference

Class SignificancesData

java.lang.Object

└-org.systemsbiology.inference.SignificancesData

public class **SignificancesData**
extends java.lang.Object

Contains a matrix of significance values of a set of observations for M elements and N evidence types. A significance of -1 means a missing observation, so no significance value exists for that (element,evidenc) ordererd pair. Non-missing significance values in this matrix are always nonnegative.

Fields

DEFAULT_SIGNIFICANCE_MISSING_DATA

public static final double **DEFAULT_SIGNIFICANCE_MISSING_DATA**

NULL_SIGNIFICANCE_STRING

public static final java.lang.String **NULL_SIGNIFICANCE_STRING**

Constructors

SignificancesData

public **SignificancesData**()

Methods

getEvidenceNames

public java.lang.String[] **getEvidenceNames**()

getNumElements

public int **getNumElements**()

getNumEvidences

public int **getNumEvidences**()

(continued from last page)

getEvidenceName

```
public java.lang.String getEvidenceName(int pColumn)
```

getElementName

```
public java.lang.String getElementName(int pRow)
```

getValueAt

```
public double getValueAt(int pRow,  
                          int pColumn)
```

getSignificancesMatrix

```
public cern.colt.matrix.DoubleMatrix2D getSignificancesMatrix()
```

getMissingDataRate

```
public double getMissingDataRate()
```

loadFromFile

```
public void loadFromFile(java.io.BufferedReader pBufferedReader,  
                        DataFileDelimiter pDelimiter)  
    throws java.io.IOException,  
           InvalidInputException
```

Package

org.systemsbiology.math

p>

org.systemsbiology.math

Class AccuracyException

```
java.lang.Object
  |
  +- java.lang.Throwable
      |
      +- java.lang.Exception
          |
          +- org.systemsbiology.math.AccuracyException
```

public class **AccuracyException**

extends java.lang.Exception

The requested operation or calculation could not be performed at the requested accuracy.

Constructors

AccuracyException

```
public AccuracyException(java.lang.String pMessage)
```

AccuracyException

```
public AccuracyException(java.lang.String pMessage,
                          java.lang.Throwable pCause)
```

org.systemsbiology.math

Class DoubleVector

java.lang.Object

|
+-org.systemsbiology.math.DoubleVector

public class **DoubleVector**extends java.lang.Object

Constructors

DoubleVector

public **DoubleVector**()

Methods

zeroNegativeElements

public static void **zeroNegativeElements**(double vec)

zeroElements

public static void **zeroElements**(double vec)

max

public static double **max**(double vec)

sumElements

public static double **sumElements**(double vec)

add

public static void **add**(double addendX,
double addendY,
double sum)

(continued from last page)

subtract

```
public static void subtract(double addendX,  
                             double addendY,  
                             double sum)
```

scalarMultiply

```
public static void scalarMultiply(double vector,  
                                    double scalar,  
                                    double product)
```

scalarMultiply

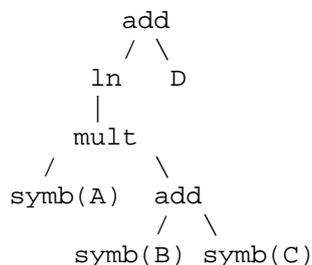
```
public static void scalarMultiply(double vector,  
                                    double scalar)
```


The following code fragment illustrates a sample use of this class:

```
MathExpression exp = new MathExpression("(A + B)/C");
System.out.println(exp.toString());
```

the above code fragment will result in $(A+B)/C$ being printed to standard output. For a more non-trivial example code using this class, please refer to the `main()` function below, which serves as a test program for this class.

When an expression string has been parsed, the result is a parse tree rooted at a single object called the "root node". As an example, the parse tree for the expression $\ln(A * (B + C)) + D$ might look (conceptually) like this:



where "mult" represents an element whose **element code** is `ElementCode.MULT`, and "add" represents an element whose element code is `ElementCode.ADD`. Furthermore, the "edges" on the above graph represent links between an element and its "operands" (i.e., its child nodes, in the tree). The "symb(A)" notation means an element whose element code is `ElementCode.SYMBOL`, and whose **symbol name** field is set to the string "A". This class is capable of parsing numeric literals in scientific notation, such as $1.0e-7$ and $2.7e+14$. The "e" may be in either lower- or upper-case. Thanks to Adam Duguid, for submitting a patch for handling the `min()` and `max()` functions.

Fields

ZERO

```
public static final org.systemsbiology.math.Expression ZERO
```

ONE

```
public static final org.systemsbiology.math.Expression ONE
```

Constructors

Expression

```
protected Expression()
```

Expression

```
public Expression(double pValue)
```

Expression

```
public Expression(java.lang.String pExpression)
```

Methods

(continued from last page)

isValidSymbol

```
public static boolean isValidSymbol(java.lang.String pToken)
```

Returns true of the string token passed as pToken is a valid "symbol", meaning that it does not contain any of the operators *, -, / +, ^, or any parentheses. In addition, the following characters may not appear in symbols because they are reserved for future use in mathematical expressions: !@#\$%[]|><{ }

Examples of valid symbols would be:

```
X1  
SY2C
```

Examples of invalid symbols would be:

```
X1-2  
X(1)  
-X  
(X)  
X*B  
X+C  
X B  
exp  
1.7
```

Parameters:

pToken - the string token to be checked for validity as a symbol

Returns:

true of the string token passed as pToken is a valid "symbol".

toString

```
public java.lang.String toString()  
    throws java.lang.IllegalStateException
```

Returns a human-comprehensible representation of this mathematical expression.

Returns:

a human-comprehensible representation of this mathematical expression.

toString

```
public java.lang.String toString(Expression.SymbolPrinter pSymbolPrinter)  
    throws java.lang.IllegalStateException,  
           DataNotFoundException
```

isFunctionName

```
public static boolean isFunctionName(java.lang.String pName)
```

Returns true if the argument is a valid function name, or false otherwise.

setExpression

```
public void setExpression(java.lang.String pExpressionString)  
    throws java.lang.IllegalArgumentException
```

Parses the mathematical expression defined by the string pExpressionString and stores the parse tree within this Expression object. This method can be called even if the Expression object was initialized with a different expression string, in which case the string specified by pExpressionString is parsed and the parse tree replaces the parse tree created when the object was constructed.

(continued from last page)

Parameters:

pExpressionString - the string definition of the mathematical expression to be parsed

computeValue

```
public double computeValue(java.util.HashMap pSymbolsMap)
    throws DataNotFoundException,
           java.lang.IllegalStateException
```

visit

```
public void visit(Expression.IVisitor pVisitor)
```

computeValue

```
public double computeValue(SymbolEvaluator pSymbolEvaluator)
    throws DataNotFoundException,
           java.lang.IllegalStateException,
           java.lang.IllegalArgumentException
```

Return the computed value of the expression (must have been defined already in the constructor, or in a call to {[@link #setExpression\(String\)](#)}), using the symbol values defined in the map ISymbolValueMap.

clone

```
public java.lang.Object clone()
```

computePartialDerivative

```
public Expression computePartialDerivative(Symbol pSymbol,
                                           SymbolEvaluator pSymbolEvaluator)
    throws DataNotFoundException
```

computePartialDerivative

```
public Expression computePartialDerivative(Symbol pSymbol,
                                           java.util.HashMap pSymbolsMap)
    throws DataNotFoundException
```

isSimpleNumber

```
public boolean isSimpleNumber()
```

getSimpleNumberValue

```
public double getSimpleNumberValue()
    throws java.lang.IllegalStateException
```

(continued from last page)

square

```
public static Expression square(Expression A)
```

multiply

```
public static Expression multiply(Expression A,  
                                   Expression B)
```

divide

```
public static Expression divide(Expression A,  
                                   Expression B)
```

negate

```
public static Expression negate(Expression A)
```

subtract

```
public static Expression subtract(Expression A,  
                                   Expression B)
```

add

```
public static Expression add(Expression A,  
                               Expression B)
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.math

Interface **Expression.IVisitor**

public interface **Expression.IVisitor**

Methods

visit

```
public void visit(Symbol pSymbol)
```

org.systemsbiology.math

Interface **Expression.SymbolPrinter**

public interface **Expression.SymbolPrinter**

Methods

printSymbol

```
public java.lang.String printSymbol(Symbol pSymbol)  
    throws DataNotFoundException
```

org.systemsbiology.math

Class MathFunctions

```
java.lang.Object
  |
  +--org.systemsbiology.math.MathFunctions
```

```
public final class MathFunctions
extends java.lang.Object
```

This class is a collection of useful mathematical functions.

Fields

LN10

```
public static final double LN10
```

Constructors

MathFunctions

```
public MathFunctions()
```

Methods

factorial

```
public static long factorial(int pArg)
```

Returns the factorial of an integer argument.

Returns:

the factorial of an integer argument.

thetaFunction

```
public static double thetaFunction(double pArg)
```

Returns 0 if the argument is negative, and 1 if the argument is nonnegative.

Returns:

0 if the argument is negative, and 1 if the argument is nonnegative.

chooseFunction

```
public static double chooseFunction(long N,
                                     int M)
    throws java.lang.IllegalArgumentException
```

This function computes N choose M , for small values of M . It is required that $N > 0$, $M \geq 0$, and $M \leq N$. **WARNING:** This implementation will generate an overflow or underflow for large values of M , but it will work for small values of M .

log10

```
public static double log10(double pArg)
```

Returns the logarithm base 10, of the argument.

stats

```
public static void stats(double pVec,  
                          MutableDouble pMean,  
                          MutableDouble pStdDev)
```

sign

```
public static double sign(double x)
```

extendedSimpsonsRule

```
public static double extendedSimpsonsRule(double pVals,  
                                             double pXmin,  
                                             double pXmax,  
                                             int pNmin,  
                                             int pNmax)
```

org.systemsbiology.math

Class MutableBoolean

java.lang.Object

└--org.systemsbiology.math.MutableBoolean

public final class **MutableBoolean**

extends java.lang.Object

A container class for a boolean native data type. This class allows you to alter the boolean variable that it contains; it is a fully mutable object. The purpose of this class is to provide a mechanism to use boolean values as values of a HashMap, while allowing those values to be mutable as well; this cannot be done with the standard Java class Boolean, which is immutable.

See Also:[MutableInteger](#), [MutableDouble](#)

Constructors

MutableBoolean

public **MutableBoolean**(boolean pBoolean)

Methods

getValue

public boolean **getValue**()

setValue

public void **setValue**(boolean pBoolean)

booleanValue

public boolean **booleanValue**()

clone

public java.lang.Object **clone**()

toString

public java.lang.String **toString**()

org.systemsbiology.math

Class MutableDouble

java.lang.Object

└--org.systemsbiology.math.MutableDouble

public final class **MutableDouble**

extends java.lang.Object

A container class for a `double` native data type. This class allows you to alter the `double` variable that it contains; it is a fully mutable object. The purpose of this class is to provide a mechanism to use `double` values as values of a `HashMap`, while allowing those values to be mutable as well; this cannot be done with the standard Java class `Double`, which is immutable.

See Also:[MutableBoolean](#), [MutableInteger](#)

Constructors

MutableDouble

public **MutableDouble**(double pDouble)

Methods

getValue

public double **getValue**()

setValue

public void **setValue**(double pDouble)

doubleValue

public double **doubleValue**()

compare

public static int **compare**(MutableDouble p1,
MutableDouble p2)

clone

public java.lang.Object **clone**()

toString

```
public java.lang.String toString()
```

org.systemsbiology.math

Class MutableInteger

java.lang.Object

```
└--org.systemsbiology.math.MutableInteger
```

```
public final class MutableInteger
extends java.lang.Object
```

A container class for a `integer` native data type. This class allows you to alter the `integer` variable that it contains; it is a fully mutable object. The purpose of this class is to provide a mechanism to use `integervalue`s as values of a `HashMap`, while allowing those values to be mutable as well; this cannot be done with the standard Java class `Integer`, which is immutable.

See Also:

[MutableBoolean](#), [MutableDouble](#)

Constructors

MutableInteger

```
public MutableInteger(int pInteger)
```

Methods

getValue

```
public int getValue()
```

setValue

```
public void setValue(int pInteger)
```

integerValue

```
public int integerValue()
```

clone

```
public java.lang.Object clone()
```

toString

```
public java.lang.String toString()
```

org.systemsbiology.math

Class **ReservedSymbolMapper**

java.lang.Object

└--org.systemsbiology.math.ReservedSymbolMapper

Direct Known Subclasses:

[ReservedSymbolMapperChemCommandLanguage](#)

public abstract class **ReservedSymbolMapper**
extends java.lang.Object

Constructors

ReservedSymbolMapper

public **ReservedSymbolMapper**()

Methods

isReservedSymbol

public abstract boolean **isReservedSymbol**(Symbol pSymbol)

getReservedSymbolValue

public abstract double **getReservedSymbolValue**(Symbol pSymbol,
SymbolEvaluator pSymbolEvaluator)
throws DataNotFoundException

getReservedSymbolNames

public abstract java.util.Collection **getReservedSymbolNames**()

org.systemsbiology.math

Class ScientificNumberFormat

```
java.lang.Object
  |
  +- java.text.Format
      |
      +- java.text.NumberFormat
          |
          +- java.text.DecimalFormat
              |
              +- org.systemsbiology.math.ScientificNumberFormat
```

```
public class ScientificNumberFormat
```

```
extends java.text.DecimalFormat
```

Truncates and formats a floating-point number based on a relative and an absolute tolerance

Constructors

ScientificNumberFormat

```
public ScientificNumberFormat(SignificantDigitsCalculator
pSignificantDigitsCalculator)
```

Methods

setNaNString

```
public void setNaNString(java.lang.String pNaNString)
```

getNaNString

```
public java.lang.String getNaNString()
```

setInfinityString

```
public void setInfinityString(java.lang.String pInfinityString)
```

getInfinityString

```
public java.lang.String getInfinityString()
```

getMinimumDigitsForScientificNotation

```
public int getMinimumDigitsForScientificNotation()
```

setMinimumDigitsForScientificNotation

```
public void setMinimumDigitsForScientificNotation(int  
pMinimumDigitsForScientificNotation)
```

format

```
public java.lang.StringBuffer format(double pValue,  
                                     java.lang.StringBuffer pResults,  
                                     java.text.FieldPosition pFieldPosition)
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.math

Class SignificantDigitsCalculator

java.lang.Object

└--org.systemsbiology.math.SignificantDigitsCalculator

```
public class SignificantDigitsCalculator
extends java.lang.Object
```

Calculates the number of significant digits in a floating-point number, based on the absolute and relative error tolerances.

Fields

SIGNIFICANT_DIGITS_UNKNOWN

```
public static final int SIGNIFICANT_DIGITS_UNKNOWN
```

Constructors

SignificantDigitsCalculator

```
public SignificantDigitsCalculator()
```

SignificantDigitsCalculator

```
public SignificantDigitsCalculator(java.lang.Double pRelTol,
                                   java.lang.Double pAbsTol)
```

Methods

setRelTol

```
public void setRelTol(java.lang.Double pRelTol)
```

setAbsTol

```
public void setAbsTol(java.lang.Double pAbsTol)
```

getRelTol

```
public java.lang.Double getRelTol()
```

(continued from last page)

getAbsTol

```
public java.lang.Double getAbsTol()
```

calculate

```
public int calculate(double pValue)
```

main

```
public final static void main(java.lang.String pArgs)
```

org.systemsbiology.math

Class Symbol

```
java.lang.Object
  |
  +--org.systemsbiology.math.Symbol
```

```
public final class Symbol
extends java.lang.Object
```

Contains a string identifier and an (optional) array index. Used by the {[@link SymbolValue](#)} class. The array index can be used instead of the string identifier, in order to find the symbol's value.

Fields

NULL_ARRAY_INDEX

```
public static final int NULL_ARRAY_INDEX
```

Constructors

Symbol

```
public Symbol(java.lang.String pSymbolName)
```

Methods

hasArrayIndex

```
public boolean hasArrayIndex()
```

getName

```
public java.lang.String getName()
```

setArrayIndex

```
public void setArrayIndex(int pArrayIndex)
```

getArrayIndex

```
public int getArrayIndex()
```

(continued from last page)

setArray

```
public void setArray(double pArray)
```

setArray

```
public void setArray(Value pArray)
```

getDoubleArray

```
public double[] getDoubleArray()
```

getValueArray

```
public Value[] getValueArray()
```

equals

```
public boolean equals(Symbol pSymbol)
```

clone

```
public java.lang.Object clone()
```

copyIndexInfo

```
public void copyIndexInfo(Symbol pSymbol)
```

clearIndexInfo

```
public void clearIndexInfo()
```

toString

```
public java.lang.String toString()
```

org.systemsbiology.math

Class SymbolEvaluationPostProcessor

java.lang.Object

└--org.systemsbiology.math.SymbolEvaluationPostProcessor

All Implemented interfaces:

java.lang.Cloneable

Direct Known Subclasses:

[SymbolEvaluationPostProcessorChemMarkupLanguage](#)

public abstract class **SymbolEvaluationPostProcessor**

extends java.lang.Object

implements java.lang.Cloneable

Constructors

SymbolEvaluationPostProcessor

public **SymbolEvaluationPostProcessor**()

Methods

modifySymbol

public abstract java.lang.String **modifySymbol**(Symbol pSymbol)
throws DataNotFoundException

modifyResult

public abstract double **modifyResult**(Symbol pSymbol,
SymbolEvaluator pSymbolEvaluator,
double pSymbolValue)
throws DataNotFoundException

clone

public abstract java.lang.Object **clone**()

org.systemsbiology.math

Class SymbolEvaluator

java.lang.Object

└--org.systemsbiology.math.SymbolEvaluator

All Implemented interfaces:

java.lang.Cloneable

Direct Known Subclasses:

[SymbolEvaluatorHashMap](#), [SymbolEvaluatorChem](#)

public abstract class **SymbolEvaluator**

extends java.lang.Object

implements java.lang.Cloneable

Fields

NULL_ARRAY_INDEX

protected static final int NULL_ARRAY_INDEX

Constructors

SymbolEvaluator

public **SymbolEvaluator**()

SymbolEvaluator

public **SymbolEvaluator**(boolean pUseExpressionValueCaching,
SymbolEvaluationPostProcessor pSymbolEvaluationPostProcessor)

Methods

getSymbolEvaluationPostProcessor

public [SymbolEvaluationPostProcessor](#) **getSymbolEvaluationPostProcessor**()

getIndexedValue

protected final double **getIndexedValue**(int pArrayIndex,
Symbol pSymbol)
throws DataNotFoundException

getExpressionValue

```
public abstract Expression getExpressionValue(Symbol pSymbol)
                                     throws DataNotFoundException
```

Returns null if the symbol corresponds to a numeric value; or returns the Expression if the symbol corresponds to an Expression; or throws an exception if the symbol is not defined.

getValue

```
public final double getValue(Symbol pSymbol)
                       throws DataNotFoundException
```

Returns the floating-point value associated with the specified {@link Symbol}.

Parameters:

pSymbol - the {@link Symbol} object for which the value is to be returned

Returns:

the floating-point value associated with the specified symbol.

hasValue

```
public abstract boolean hasValue(Symbol pSymbol)
```

Returns true if the object has a {@link Value} defined, or false otherwise.

Returns:

true if the object has a {@link Value} defined, or false otherwise.

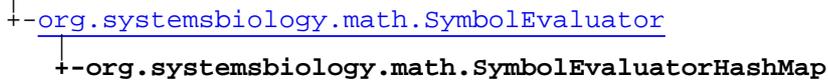
getUnindexedValue

```
public abstract double getUnindexedValue(Symbol pSymbol)
                                     throws DataNotFoundException
```

org.systemsbiology.math

Class SymbolEvaluatorHashMap

java.lang.Object

public class **SymbolEvaluatorHashMap**extends [SymbolEvaluator](#)

An implementation of the [{@link SymbolEvaluator}](#) abstract class based on a `HashMap`. The string name in the [{@link Symbol}](#) object is used as the key into the `HashMap`, and the associated [{@link Value}](#) object is the value in the map.

Fields

mSymbolMap

protected final java.util.HashMap **mSymbolMap**

Constructors

SymbolEvaluatorHashMap

public **SymbolEvaluatorHashMap**(java.util.HashMap pSymbolMap)

Methods

getValue

protected double **getValue**(java.lang.String pSymbolName)
throws `DataNotFoundException`

getUnindexedValue

public double **getUnindexedValue**(Symbol pSymbol)
throws `DataNotFoundException`

hasValue

public boolean **hasValue**(Symbol pSymbol)

getExpressionValue

public [Expression](#) **getExpressionValue**(Symbol pSymbol)
throws `DataNotFoundException`

(continued from last page)

org.systemsbiology.math Class SymbolValue

```
java.lang.Object
  |
  +--org.systemsbiology.math.SymbolValue
```

All Implemented interfaces:
java.lang.Comparable

Direct Known Subclasses:
[Reaction](#), [Parameter](#), [Species](#), [Compartment](#)

```
public class SymbolValue
  extends java.lang.Object
  implements java.lang.Comparable
```

Represents a { @link Symbol} and an associated { @link Value}. It is the base class for many other classes such as { @link org.systemsbiology.chem.Species}.

Fields

mValue

```
protected org.systemsbiology.math.Value mValue
```

mSymbol

```
protected final org.systemsbiology.math.Symbol mSymbol
```

Constructors

SymbolValue

```
public SymbolValue(java.lang.String pSymbolName)
```

Constructs a SymbolValue using the specified string symbol name. The { @link Value}. object is set to null.

SymbolValue

```
public SymbolValue(java.lang.String pSymbolName,
                  double pSymbolValue)
```

SymbolValue

```
public SymbolValue(SymbolValue pSymbolValue)
```

The copy constructor.

(continued from last page)

SymbolValue

```
public SymbolValue(java.lang.String pSymbolName,  
                  Value pValue)
```

Constructs a SymbolValue using the specified string symbol name, and the specified {@link Value} object.

Methods

addSymbolToMap

```
public final void addSymbolToMap(java.util.HashMap pMap,  
                                 java.lang.String pSymbolName)  
    throws java.lang.IllegalStateException
```

Adds itself to the specified HashMap, using the string symbol name specified as pSymbolName.

addSymbolToMap

```
public final void addSymbolToMap(java.util.HashMap pMap,  
                                 java.lang.String pSymbolName,  
                                 ReservedSymbolMapper pReservedSymbolMapper)  
    throws java.lang.IllegalArgumentException,  
           java.lang.IllegalStateException
```

equals

```
public boolean equals(SymbolValue pSymbolValue)
```

getValue

```
public final Value getValue()
```

Accessor for the {@link Value} object stored in this object.

setValue

```
public void setValue(Value pValue)
```

getSymbol

```
public Symbol getSymbol()
```

Accessor for the {@link Symbol} object stored in this object.

compareTo

```
public int compareTo(java.lang.Object pObject)
```

clone

```
public java.lang.Object clone()
```

org.systemsbiology.math

Class Value

```
java.lang.Object
  |
  +--org.systemsbiology.math.Value
```

public final class **Value**
extends java.lang.Object

Class that can represent a floating point value, or an expression representing a floating point value. An object of this class always contains either an { @link Expression } object or a { @link MutableDouble } object.

Constructors

Value

```
public Value(Expression pExpressionValue)
    Constructs a { @link Value } composed of the specified { @link Expression }.
```

Value

```
public Value(double pNumericValue)
    Constructs a { @link Value } composed of the specified floating-point value.
```

Methods

getExpressionValue

```
public Expression getExpressionValue()
```

setValue

```
public void setValue(double pValue)
    throws java.lang.IllegalStateException
    Stores the specified floating-point value.
```

setValue

```
public void setValue(Expression pExpressionValue)
    throws java.lang.IllegalArgumentException
    Stores the specified { @link Expression }. If this object was constructed using a floating-point value, an
    IllegalStateException is thrown.
```

getValue

```
public double getValue()
    throws java.lang.IllegalStateException
    Returns the floating-point value defined for this object. If the object instead has an { @link Expression } stored within it, an
    IllegalStateException is thrown.
```

isExpression

```
public boolean isExpression()
```

Returns true if the object has an {@link Expression} object stored within it, or false otherwise.

getValueWithCaching

```
public double getValueWithCaching(SymbolEvaluator pSymbolValueMap)
    throws DataNotFoundException
```

If this object contains an {@link Expression}, computes the value of the Expression using the supplied {@link SymbolEvaluator}; otherwise it returns the floating-point value stored in the internal MutableDouble object within this object.

getValue

```
public double getValue(SymbolEvaluator pSymbolValueMap)
    throws DataNotFoundException
```

getExpressionString

```
public java.lang.String getExpressionString()
    throws java.lang.IllegalStateException
```

Return a string representation of the {@link Expression} object stored in this class. If no expression object is stored within this class, an IllegalStateException is thrown.

getExpressionString

```
public java.lang.String getExpressionString(Expression.SymbolPrinter pSymbolPrinter)
    throws java.lang.IllegalStateException,
           DataNotFoundException
```

toString

```
public java.lang.String toString()
```

clearExpressionValueCache

```
public void clearExpressionValueCache()
```

clone

```
public java.lang.Object clone()
```

Package

org.systemsbiology.math.probability

/p>

org.systemsbiology.math.probability

Class **DistributionPrinter**

java.lang.Object

└-org.systemsbiology.math.probability.DistributionPrinter

public class **DistributionPrinter**

extends java.lang.Object

Creates a string summary of a probability distribution function.

Constructors

DistributionPrinter

public **DistributionPrinter**()

Methods

print

public static java.lang.String **print**(IContinuousDistribution pDist,
int pNumSamples)

org.systemsbiology.math.probability

Class Empirical

java.lang.Object

└--org.systemsbiology.math.probability.Empirical

All Implemented interfaces:

[IContinuousDistribution](#)

public class **Empirical**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

Empirical

```
public Empirical(cern.colt.matrix.DoubleMatrix1D pDist,  
                 double pMin,  
                 double pMax)
```

Methods

cdf

```
public double cdf(double x)
```

pdf

```
public double pdf(double x)
```

mean

```
public double mean()
```

domainMin

```
public double domainMin()
```

domainMax

```
public double domainMax()
```

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class Gamma

java.lang.Object

└--org.systemsbiology.math.probability.Gamma

All Implemented interfaces:

[IContinuousDistribution](#)

public class **Gamma**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

Gamma

```
public Gamma(double pMean,  
             double pVariance)
```

Methods

mean

```
public double mean()
```

pdf

```
public double pdf(double x)
```

cdf

```
public double cdf(double x)
```

domainMin

```
public double domainMin()
```

domainMax

```
public double domainMax()
```

(continued from last page)

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class HalfLorentz

java.lang.Object

└--org.systemsbiology.math.probability.HalfLorentz

All Implemented interfaces:

[IContinuousDistribution](#)

public class **HalfLorentz**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

HalfLorentz

public **HalfLorentz**(double pStdev)

Methods

pdf

public double **pdf**(double x)

cdf

public double **cdf**(double x)

domainMin

public double **domainMin**()

domainMax

public double **domainMax**()

mean

public double **mean**()

(continued from last page)

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class HalfNormal

java.lang.Object

└--org.systemsbiology.math.probability.HalfNormal

All Implemented interfaces:

[IContinuousDistribution](#)

public class **HalfNormal**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

HalfNormal

public **HalfNormal**(double pMean)

Methods

pdf

public double **pdf**(double x)

cdf

public double **cdf**(double x)

domainMin

public double **domainMin**()

domainMax

public double **domainMax**()

mean

public double **mean**()

(continued from last page)

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Interface IContinuousDistribution

public interface **IContinuousDistribution**

Methods

pdf

```
public double pdf(double x)
```

cdf

```
public double cdf(double x)
```

domainMin

```
public double domainMin()
```

domainMax

```
public double domainMax()
```

mean

```
public double mean()
```

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class Laplace

java.lang.Object

└--org.systemsbiology.math.probability.Laplace

All Implemented interfaces:

[IContinuousDistribution](#)

public class **Laplace**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

Laplace

```
public Laplace(double pMean,  
               double pVariance)
```

Methods

pdf

```
public double pdf(double x)
```

cdf

```
public double cdf(double x)
```

domainMin

```
public double domainMin()
```

domainMax

```
public double domainMax()
```

mean

```
public double mean()
```

(continued from last page)

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class Logistic

java.lang.Object

└--org.systemsbiology.math.probability.Logistic

All Implemented interfaces:

[IContinuousDistribution](#)

public class **Logistic**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

Logistic

```
public Logistic(double pMean,  
                double pVariance)
```

Methods

pdf

```
public double pdf(double x)
```

cdf

```
public double cdf(double x)
```

domainMin

```
public double domainMin()
```

domainMax

```
public double domainMax()
```

mean

```
public double mean()
```

(continued from last page)

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class Lorentz

java.lang.Object

└--org.systemsbiology.math.probability.Lorentz

All Implemented interfaces:

[IContinuousDistribution](#)

public class **Lorentz**

extends java.lang.Object

implements [IContinuousDistribution](#)

Fields

RATIO_WIDTH_TO_STDEV

public static final double **RATIO_WIDTH_TO_STDEV**

Constructors

Lorentz

public **Lorentz**(double pMean,
double pStdev)

Methods

pdf

public static double **pdf**(double pMean,
double pWidth,
double x)

cdf

public static double **cdf**(double pMean,
double pWidth,
double x)

pdf

public double **pdf**(double x)

(continued from last page)

cdf

```
public double cdf(double x)
```

domainMin

```
public double domainMin()
```

domainMax

```
public double domainMax()
```

mean

```
public double mean()
```

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class Maxwell

java.lang.Object

└--org.systemsbiology.math.probability.Maxwell

All Implemented interfaces:

[IContinuousDistribution](#)public class **Maxwell**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

Maxwell

public **Maxwell**(double pMean)

Methods

cdf

public double **cdf**(double x)

pdf

public double **pdf**(double x)

domainMin

public double **domainMin**()

domainMax

public double **domainMax**()

mean

public double **mean**()

(continued from last page)

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class Normal

java.lang.Object

└-org.systemsbiology.math.probability.Normal

All Implemented interfaces:

[IContinuousDistribution](#)

public class **Normal**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

Normal

```
public Normal(double pMean,  
              double pVariance)
```

Methods

pdf

```
public static double pdf(double pMean,  
                        double pVariance,  
                        double x)
```

cdf

```
public static double cdf(double pMean,  
                        double pVariance,  
                        double x)
```

pdf

```
public double pdf(double x)
```

cdf

```
public double cdf(double x)
```

domainMin

```
public double domainMin()
```

(continued from last page)

domainMax

```
public double domainMax()
```

mean

```
public double mean()
```

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

org.systemsbiology.math.probability

Class Rayleigh

java.lang.Object

└-org.systemsbiology.math.probability.Rayleigh

All Implemented interfaces:

[IContinuousDistribution](#)public class **Rayleigh**

extends java.lang.Object

implements [IContinuousDistribution](#)

Constructors

Rayleigh

public **Rayleigh**(double pMean)

Methods

pdf

public double **pdf**(double x)

cdf

public double **cdf**(double x)

domainMin

public double **domainMin**()

domainMax

public double **domainMax**()

mean

public double **mean**()

(continued from last page)

variance

```
public double variance()
```

name

```
public java.lang.String name()
```

Package

org.systemsbiology.util

p>

org.systemsbiology.util
Class AppConfig

```
java.lang.Object
  |
  +--org.systemsbiology.util.AppConfig
```

```
public class AppConfig
  extends java.lang.Object
```

Utility class for reading an XML configuration file for an application. Provides standard methods for obtaining the application name, version, release date, and home page. This class can be subclassed to add new properties.

Fields

CONFIG_FILE_NAME

```
public static final java.lang.String CONFIG_FILE_NAME
```

Constructors

AppConfig

```
public AppConfig(java.io.File pConfigFile)
  Creates an AppConfig using the config file pConfigFile.
```

AppConfig

```
public AppConfig(java.lang.Class pAppClass)
  Creates an AppConfig using class pAppClass to search for the config file resource. That class must have a resource file AppConfig.xml.
```

Methods

getProperty

```
public java.lang.String getProperty(java.lang.String pPropertyName)
```

getAppHomePage

```
public java.lang.String getAppHomePage()
```

getAppName

```
public java.lang.String getAppName()
```

(continued from last page)

getAppVersion

```
public java.lang.String getAppVersion()
```

getAppDate

```
public java.lang.String getAppDate()
```

getAppIconURL

```
public java.lang.String getAppIconURL()
```

getAppHelpSetName

```
public java.lang.String getAppHelpSetName()
```

getAppMaintainerFullName

```
public java.lang.String getAppMaintainerFullName()
```

getAppCopyright

```
public java.lang.String getAppCopyright()
```

get

```
public static AppConfig get(java.lang.Class pAppClass,  
                             java.lang.String pAppDir)  
    throws DataNotFoundException,  
           InvalidInputException,  
           java.io.FileNotFoundException
```

main

```
public static void main(java.lang.String pArgs)
```

org.systemsbiology.util Class ClassRegistry

```
java.lang.Object
  |
  +--org.systemsbiology.util.ClassRegistry
```

```
public class ClassRegistry
extends java.lang.Object
```

Implements a class registry for a given interface (that must itself extend the {@link IAliasableClass} marker interface). This registry is capable of searching the entire java CLASSPATH for all classes that implement the specified interface. A hash of instances of the given interface is also stored, so that an instance of any class implementing the interface, can be retrieved by referring to the `"class alias"`.

Constructors

ClassRegistry

```
public ClassRegistry(java.lang.Class pInterface)
```

Create a ClassRegistry instance. The pInterface argument must specify an interface that extends the IAliasableClass interface. Lets assume the interface is called `"IFoo"`. The class registry instance (when you call buildRegistry will build a list of all objects in the classpath that implement the IFoo interface, and that contain the CLASS_ALIAS public static String field.

Methods

buildRegistry

```
public void buildRegistry()
    throws java.lang.ClassNotFoundException,
           java.io.IOException,
           java.lang.IllegalArgumentException
```

Searches the classpath for all classes implementing the interface that was specified in the constructor. This method will take a while to complete, because it is searching the entire classpath.

getClass

```
public java.lang.Class getClass(java.lang.String pClassAlias)
    throws DataNotFoundException
```

Returns the Class object associated with the specified class alias (a string identifier that uniquely identifies a particular implementation of the interface that was passed to the constructor for this ClassRegistry object). If no class is found corresponding to the specified alias, an exception is thrown.

Parameters:

pClassAlias - the alias of the class that is to be returned

Returns:

the Class object associated with the specified class alias

getInstance

```
public java.lang.Object getInstance(java.lang.String pClassAlias)
    throws DataNotFoundException
```

(continued from last page)

Returns an instance of the class corresponding to the specified class alias `pClassAlias`. This object will be an instance of a class that implements the interface that was passed to the `ClassRegistry` constructor. If no such instance exists, but the class corresponding to the alias is known, an instance will be created and the reference will be stored and returned.

Parameters:

`pClassAlias` - the alias of the class that is to be returned

Returns:

an instance of the class corresponding to the specified class alias `pClassAlias`

printRegistry

```
public void printRegistry(java.io.PrintStream pStream)
```

Print out a summary of the contents of the class registry, to the specified `PrintStream pString`.

getRegistryAliasesCopy

```
public java.util.Set getRegistryAliasesCopy()
```

Return a `Set` containing copies of all of the aliases (as strings) for objects implementing the interface that was passed to the `ClassRegistry` constructor.

Returns:

a `Set` containing copies of all of the aliases (as strings) for objects implementing the interface that was passed to the `ClassRegistry` constructor.

clearInstances

```
public void clearInstances()
```

main

```
public static void main(java.lang.String pArgs)
```

Test method for this class

org.systemsbiology.util

Class CommandLineApp

java.lang.Object

└--org.systemsbiology.util.CommandLineApp

Direct Known Subclasses:

[SimulationLauncherCommandLine](#)

```
public abstract class CommandLineApp
extends java.lang.Object
```

Class template for a commane-line application

Fields

HELP_ARG

protected static final java.lang.String **HELP_ARG**

NULL_MODIFIER

public static final java.lang.String **NULL_MODIFIER**

Constructors

CommandLineApp

public **CommandLineApp**()

Methods

printUsage

protected abstract void **printUsage**(java.io.OutputStream pOutputStream)

handleCommandLineError

protected void **handleCommandLineError**(java.lang.String pMessage)

getRequiredArgumentModifier

protected java.lang.String **getRequiredArgumentModifier**(java.lang.String pArgument,
java.lang.String pArgs,
int pCtr)

(continued from last page)

getRequiredDoubleArgumentModifier

```
protected java.lang.Double getRequiredDoubleArgumentModifier(java.lang.String
pArgument,
                                                                java.lang.String pArgs,
                                                                int pCtr)
```

getRequiredIntegerArgumentModifier

```
protected java.lang.Integer getRequiredIntegerArgumentModifier(java.lang.String
pArgument,
                                                                java.lang.String pArgs,
                                                                int pCtr)
```

getRequiredLongArgumentModifier

```
protected java.lang.Long getRequiredLongArgumentModifier(java.lang.String pArgument,
                                                            java.lang.String pArgs,
                                                            int pCtr)
```

checkAndHandleHelpRequest

```
protected void checkAndHandleHelpRequest(java.lang.String pArgs)
```

handleCommandLine

```
protected abstract void handleCommandLine(java.lang.String pArgs)
```

org.systemsbiology.util

Class DataNotFoundException

```
java.lang.Object
  |
  +- java.lang.Throwable
      |
      +- java.lang.Exception
          |
          +- org.systemsbiology.util.DataNotFoundException
```

```
public class DataNotFoundException
extends java.lang.Exception
```

Represents an error condition in which the caller has requested a particular data element, but the data element was not found.

Constructors

DataNotFoundException

```
public DataNotFoundException(java.lang.String pMessage)
```

DataNotFoundException

```
public DataNotFoundException(java.lang.String pMessage,
                             java.lang.Throwable pCause)
```

org.systemsbiology.util
Class DebugUtils

java.lang.Object
└--org.systemsbiology.util.DebugUtils

public class **DebugUtils**
extends java.lang.Object

Constructors

DebugUtils

public **DebugUtils**()

Methods

setDebug

public static void **setDebug**(boolean pDebug)

getDebug

public static boolean **getDebug**()

printDoubleVector

public static void **printDoubleVector**(double pVec)

describeSortedObjectList

public static void **describeSortedObjectList**(java.lang.StringBuffer pStringBuffer,
java.util.HashMap pObjectMap,
java.lang.Class pClassTypeFilter)

describeSortedObjectList

public static void **describeSortedObjectList**(java.lang.StringBuffer pStringBuffer,
java.util.HashMap pObjectMap,
java.lang.String pSeparatorString)

(continued from last page)

describeSortedObjectList

```
public static void describeSortedObjectList(java.lang.StringBuffer pStringBuffer,  
                                             java.util.HashMap pObjectMap)
```

describeSortedObjectList

```
public static void describeSortedObjectList(java.lang.StringBuffer pStringBuffer,  
                                             java.util.HashMap pObjectMap,  
                                             java.lang.Class pClassTypeFilter,  
                                             java.lang.String pSeparatorString)
```

org.systemsbiology.util

Class **ExceptionUtils**

java.lang.Object

└─org.systemsbiology.util.**ExceptionUtils**

public class **ExceptionUtils**

extends java.lang.Object

Provides utility methods related to handling exceptions.

Constructors

ExceptionUtils

public **ExceptionUtils**()

Methods

getStackTrace

public static java.lang.String **getStackTrace**(java.lang.Throwable pThrowable)

Returns the stack backtrace of an exception, as a string.

Parameters:

pThrowable - the exception whose stack backtrace is to be returned

Returns:

the stack backtrace of an exception, as a string.

org.systemsbiology.util

Class FileUtils

java.lang.Object

└─org.systemsbiology.util.FileUtils

public class **FileUtils**
extends java.lang.Object

Utility class for manipulating file names

Constructors

FileUtils

public **FileUtils**()

Methods

addSuffixToFilename

public static java.lang.String **addSuffixToFilename**(java.lang.String pFileName,
java.lang.String pSuffix)

getExtension

public static java.lang.String **getExtension**(java.lang.String pFileName)

removeExtension

public static java.lang.String **removeExtension**(java.lang.String pFileName)

fixWindowsCommandLineDirectoryNameMangling

public static java.lang.String **fixWindowsCommandLineDirectoryNameMangling**
(java.lang.String pDirName)

The Windows command-line sometimes passes a file name terminating in a double-quote character. This method strips out the double-quote character and appends a directory separator character, if necessary.

createFileURL

public static java.lang.String **createFileURL**(java.io.File pFile)

org.systemsbiology.util

Interface IAliasableClass

public interface **IAliasableClass**

Marker interface defining a class that contains a "CLASS_ALIAS" field of type String (public). The value of this field is the "alias" for the class, a short name used by the class registry built in the {[@link ClassRegistry](#)} class.

Fields

CLASS_ALIAS

```
public static final java.lang.String CLASS_ALIAS
```

org.systemsbiology.util

Class IncludeHandler

java.lang.Object

└--org.systemsbiology.util.IncludeHandler

```
public class IncludeHandler
extends java.lang.Object
```

Utility class for handling nested file inclusion, in a scripting environment.

Constructors

IncludeHandler

```
public IncludeHandler()
```

Methods

getDirectory

```
public java.io.File getDirectory()
```

setDirectory

```
public void setDirectory(java.io.File pDirectory)
    throws java.lang.IllegalArgumentException
```

getIncludeFileAbsolutePath

```
public java.lang.String getIncludeFileAbsolutePath(java.lang.String pIncludeFileName)
    throws java.io.IOException
```

openReaderForIncludeFile

```
public java.io.BufferedReader openReaderForIncludeFile(java.lang.String
pIncludedFileName,
    java.nio.charset.Charset
pCharset)
    throws java.io.IOException
```

isWithinIncludeFile

```
public boolean isWithinIncludeFile()
```

org.systemsbiology.util

Class **InvalidInputException**

```
java.lang.Object
  |
  +- java.lang.Throwable
      |
      +- java.lang.Exception
          |
          +- org.systemsbiology.util.InvalidInputException
```

```
public class InvalidInputException
extends java.lang.Exception
```

Constructors

InvalidInputException

```
public InvalidInputException(java.lang.String pMessage)
```

InvalidInputException

```
public InvalidInputException(java.lang.String pMessage,
                             java.lang.Throwable pCause)
```

Index

A

AboutDialog 103
AbstractComparator 126
accept 104
accept 154
AccuracyException 196
add 130
add 131
add 135
add 138
add 197
add 203
addApproveListener 152
addListener 117
addParameter 14
addParameter 25
addProduct 27
addReactant 27
addReaction 14
addReactionParticipantToMap 27
addRequestedSymbolValues 45
addRow 132
addSpecies 14
addSpecies 28
addSuffixToFilename 262
addSymbolToMap 225
allowsInterrupt 12
allowsInterrupt 98
allowsInterrupt 100
allowsInterrupt 50
allowsInterrupt 60
allowsNonpositiveArgument 161
ALPHA_VALUE_DEVIATION_WARNING_THRESHOLD
169
AppConfig 252
ASCENDING 156
AVOGADRO_CONSTANT 5

B

booleanValue 208
buildFromLineBasedStringDelimitedInput 133

buildModel 8
buildModel 16
buildModel 87
buildRegistry 254

C

calculate 216
calculateSignificances 189
calculateSignificancesNonParametric 188
calculateSignificancesParametric 189
canComputeFluctuations 12
canComputeFluctuations 44
canImport 151
CATEGORY_NAME 80
cdf 230
cdf 232
cdf 234
cdf 236
cdf 238
cdf 239
cdf 241
cdf 243
cdf 245
cdf 247
cdf 249
CDF_NONPARAMETRIC 185
CDF_PARAMETRIC 186
checkAndHandleHelpRequest 257
checkDoesSystemClipboardSupportImageTransfer 151
checkDynamicalSymbolsValues 58
checkIntegrity 134
checkSimulationParameters 11
checkSimulationParameters 46
checkSimulationParametersForDeterministicSimulator 47
checkSimulationParametersImpl 97
checkSimulationParametersImpl 46
checkSimulationParametersImpl 52
checkSimulationParametersImpl 54
checkSimulationParametersImpl 60
checkSimulationParametersImpl 67
ChemFileFilter 104
chooseDeltaTimeToNextReaction 59
chooseFunction 206
chooseIndexOfNextReaction 59

CLASS_ALIAS 114
CLASS_ALIAS 115
CLASS_ALIAS 16
CLASS_ALIAS 18
CLASS_ALIAS 20
CLASS_ALIAS 21
CLASS_ALIAS 98
CLASS_ALIAS 100
CLASS_ALIAS 87
CLASS_ALIAS 89
CLASS_ALIAS 51
CLASS_ALIAS 53
CLASS_ALIAS 61
CLASS_ALIAS 63
CLASS_ALIAS 65
CLASS_ALIAS 68
CLASS_ALIAS 69
CLASS_ALIAS 263
ClassRegistry 254
clear 129
clear 132
clear 135
clear 139
clearExpressionValueCache 227
clearExpressionValueCaches 46
clearIndexInfo 218
clearInstances 255
clearSimulatorState 45
clearTreeLinks 136
clone 4
clone 24
clone 25
clone 93
clone 71
clone 179
clone 202
clone 208
clone 209
clone 211
clone 218
clone 219
clone 225
clone 227
close 105
CODE_ASCENDING 156
CODE_CDF_NONPARAMETRIC 185
CODE_CDF_PARAMETRIC 185
CODE_DESCENDING 156
CODE_LINEAR 176
CODE_NONE 156
CODE_PDF_NONPARAMETRIC 185
CODE_PDF_PARAMETRIC 185
CODE_POWER 176
CODE_UNIFORM 176
COLUMN_NAME_ELEMENT 181
COMMA 127
CommandLineApp 256
compare 126
compare 209
compareTo 29
compareTo 127
compareTo 177
compareTo 186
compareTo 225
Compartment 3
ComponentUtils 142
computeDerivative 46
computeJacobian 72
computeLeapTime 66
computeLeapTime 68
computeLeapTime 69
computePartialDerivative 202
computeRate 7
computeRatePartialDerivativeExpression 47
computeReactionProbabilities 46
computeScale 49
computeSignificanceNonParametric 188
computeValue 202
CONFIG_FILE_NAME 252
Constants 5
constructDataSelector 143
constructSpeciesArrays 26
containsDelayedOrMultistepReaction 15
copyIndexInfo 218
createFileURL 262
createLabel 40
createNavigator 146
createRequestedSymbolArray 46
createSimulationResults 45
createTimesArray 46

- createTransferable 151
CSV_EXCEL 75
CSV_GNUPLOT 75
CSV_MATLAB 75
- ## D
- DataColumnSelector 143
DataManagerDriver 159
DataNormalizer 162
DataNormalizerDriver 164
DataNormalizerParams 165
DataNormalizerResults 166
DataNotFoundException 258
DebugUtils 259
DEFAULT_DATA_FILE_DELIMITER 169
DEFAULT_ENSEMBLE_SIZE 57
DEFAULT_EVIDENCE_WEIGHT_TYPE 169
DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIO
NS 95
DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIO
NS 48
DEFAULT_FLAG_GET_FINAL_SYMBOL_FLUCTUATIO
NS 57
DEFAULT_INITIAL_SIGNIFICANCE_CUTOFF 169
DEFAULT_MAX_ALLOWED_ABSOLUTE_ERROR 95
DEFAULT_MAX_ALLOWED_ABSOLUTE_ERROR 48
DEFAULT_MAX_ALLOWED_RELATIVE_ERROR 95
DEFAULT_MAX_ALLOWED_RELATIVE_ERROR 48
DEFAULT_MIN_NUM_MILLISECONDS_FOR_UPDATE
41
DEFAULT_MISSING_DATA_SIGNIFICANCE 188
DEFAULT_NUM_BINS 169
DEFAULT_NUM_HISTORY_BINS 6
DEFAULT_NUM_HISTORY_BINS 95
DEFAULT_NUM_HISTORY_BINS 48
DEFAULT_NUM_HISTORY_BINS 57
DEFAULT_QUANTILE_THRESHOLD 169
DEFAULT_REACTION_PARAMETER_SYMBOL_NAME
89
DEFAULT_SEPARATION_THRESHOLD 169
DEFAULT_SIGNIFICANCE_MISSING_DATA 193
DEFAULT_SPECIFICATION 89
DEFAULT_VOLUME 3
DelayedReactionSolver 6
DESCENDING 156
describeSortedObjectList 260
disableDoubleClick 142
displayHelpBrowser 149
DistributionPrinter 229
divide 203
doAnalysis 79
doAnalysis 81
doAnalysisCMDL 81
domainMax 230
domainMax 232
domainMax 234
domainMax 236
domainMax 238
domainMax 239
domainMax 241
domainMax 244
domainMax 245
domainMax 248
domainMax 249
domainMin 230
domainMin 232
domainMin 234
domainMin 236
domainMin 238
domainMin 239
domainMin 241
domainMin 244
domainMin 245
domainMin 247
domainMin 249
doubleValue 209
DoubleVector 197
- ## E
- EDIT 109
EDIT_COPY 110
EDIT_CUT 110
EDIT_PASTE 110
EditorPane 105
Empirical 230
EmptyTableModel 144
equals 4
equals 26

equals 29
equals 71
equals 218
equals 225
estimateSpeciesFluctuations 72
EvidenceWeightedInferer 167
EvidenceWeightedInfererDriver 170
EvidenceWeightedInfererParams 171
EvidenceWeightedInfererResults 175
ExceptionNotificationOptionPane 145
ExceptionUtils 261
ExHelpSearchNavigator 146
export 9
export 18
export 20
export 21
export 90
exportModel 113
Expression 200
extendedSimpsonsRule 207

F

f 96
factorial 206
FAILED_CLOSED 120
FAILED_RUNNING 120
FILE 109
FILE_CLOSE 110
FILE_OPEN 110
FILE_QUIT 110
FILE_SAVE 110
FILE_SAVE_AS 110
FileChooser 147
FileUtils 262
findAffectedElements 168
fixWindowsCommandLineDirectoryNameMangling 262
flush 184
format 214
FramePlacer 148

G

g 96
Gamma 232
generateChart 123
get 109
get 111
get 92
get 75
get 128
get 129
get 160
get 161
get 177
get 186
get 253
getAbsTol 215
getAlias 11
getAlias 98
getAlias 100
getAlias 45
getAlias 52
getAlias 54
getAlias 62
getAlias 64
getAlias 66
getAll 128
getAll 157
getAll 160
getAll 161
getAll 177
getAll 186
getApp 107
getAppCopyright 253
getAppDate 253
getAppHelpSetName 253
getAppHomePage 252
getAppIconURL 253
getAppMaintainerFullName 253
getAppName 252
getAppVersion 252
getArrayIndex 217
getAverageValue 140
getBufferedReader 16
getBuiltinFunctionInfo 86
getBuiltinFunctions 86
getCancelled 34
getClass 254
getCode 156

getCode 176
getCode 186
getColumn 132
getColumn 180
getColumnCount 133
getColumnCount 144
getColumnCount 182
getColumnName 182
getCombinedSignificanceCutoff 172
getCommentChar 76
getCompartment 70
getComputeFluctuations 56
getConfig 80
getConversionToMolecules 92
getConvertSubstanceToMolecules 93
getDebug 259
getDefaultExtension 128
getDefaultSimulatorParameters 12
getDefaultSimulatorParameters 97
getDefaultSimulatorParameters 50
getDefaultSimulatorParameters 60
getDelay 6
getDelay 26
getDelayedReactionEstimatedAverageFutureRate 46
getDelimiter 127
getDescription 104
getDescription 154
getDirectory 264
getDoubleArray 218
getDynamic 29
getDynamicSymbols 14
getElementName 179
getElementName 194
getElementNames 179
getElementSortStatus 182
getEndTime 39
getEnsembleSize 56
getEvidenceName 179
getEvidenceName 193
getEvidenceNames 179
getEvidenceNames 193
getEvidenceSortStatus 182
getEvidenceWeightType 172
getExpressionString 227
getExpressionValue 73
getExpressionValue 221
getExpressionValue 222
getExpressionValue 226
getExtension 262
getFileRegex 8
getFileRegex 9
getFileRegex 17
getFileRegex 18
getFileRegex 20
getFileRegex 21
getFileRegex 88
getFileRegex 90
getFilterRegex 128
getFractionComplete 37
getFractionToRemove 172
getIconByName 150
getIncludeFileAbsolutePath 264
getIndexedValue 220
getInfinity 76
getInfinityString 213
getInitialSignificanceCutoff 171
getInstance 254
getIntermedSpecies 7
getIterationCounter 36
getKineticLaw 85
getLastTimePoint 139
getLevel 91
getListener 112
getMaxAllowedAbsoluteError 56
getMaxAllowedRelativeError 56
getMaxIterations 171
getMaxReducedChiSquare 192
getMaxStepSize 53
getMinDelayedReactionDelay 47
getMinFractionalCostChange 172
getMinimumDigitsForScientificNotation 213
getMinTime 140
getMissingDataRate 179
getMissingDataRate 194
getModelName 83
getModelName 38
getModelViewerRegistry 107
getName 109
getName 111
getName 4

getName 14
getName 23
getName 27
getName 70
getName 128
getName 157
getName 160
getName 161
getName 177
getName 186
getName 217
getNaN 75
getNaNString 213
getNext 129
getNext 131
getNext 135
getNext 138
getNextDelayedReactionIndex 58
getNextResults 118
getNextSortStatusInCycle 157
getNonMissingColumnVals 180
getNthBoundarySpeciesCompartmentName 84
getNthBoundarySpeciesName 84
getNthCompartmentName 84
getNthFloatingSpeciesCompartmentName 84
getNthFloatingSpeciesName 84
getNthGlobalParameterName 85
getNthParameterHasValue 85
getNthParameterName 85
getNthParameterValue 85
getNthProductName 85
getNthProductStoichiometry 85
getNthReactantName 84
getNthReactantStoichiometry 85
getNthReactionName 84
getNthRuleFormula 86
getNthRuleName 86
getNthRuleType 86
getNumBins 171
getNumBins 192
getNumBoundarySpecies 83
getNumCompartments 83
getNumElements 179
getNumElements 193
getNumEvidences 179
getNumEvidences 193
getNumFloatingSpecies 83
getNumGlobalParameters 84
getNumHistoryBins 7
getNumHistoryBins 55
getNumLocalSymbols 26
getNumObservations 180
getNumParameters 85
getNumParticipants 26
getNumProducts 26
getNumProducts 84
getNumReactants 26
getNumReactants 84
getNumReactions 83
getNumRules 86
getNumSteps 26
getNumStoredPoints 140
getOrderedResultsSymbolNamesArray 15
getOrderedSpeciesNamesArray 15
getParameters 25
getParserAliasesList 121
getPoissonEvent 59
getPreference 184
getProperty 252
getRandomNumberGenerator 58
getRandomNumberUniformInterval 58
getRate 6
getReactionRateExpressions 47
getReactions 14
getRegistryAliasesCopy 255
getRelTol 215
getRequiredArgumentModifier 256
getRequiredDoubleArgumentModifier 257
getRequiredIntegerArgumentModifier 257
getRequiredLongArgumentModifier 257
getReservedSymbolMapper 13
getReservedSymbolMapper 73
getReservedSymbolNames 32
getReservedSymbolNames 212
getReservedSymbolValue 32
getReservedSymbolValue 212
getResultsDateTime 40
getResultsFinalSymbolFluctuations 38
getResultsSymbolNames 39
getResultsSymbolValues 39

-
- getResultsTimeValues 39
 - getRow 132
 - getRowCount 132
 - getRowCount 144
 - getRowCount 182
 - getScreenSize 148
 - getSelectedColumns 152
 - getSignificanceCalculationMethod 191
 - getSignificancesMatrix 194
 - getSimpleNumberValue 202
 - getSimulationFinished 36
 - getSimulatorAlias 38
 - getSimulatorAliasesCopy 117
 - getSimulatorAliasesList 121
 - getSimulatorParameters 39
 - getSingle 127
 - getSingleTailed 191
 - getSmoothingLength 172
 - getSmoothingLength 192
 - getSortedFileFormatNames 76
 - getSourceActions 151
 - getSpecies 29
 - getSpeciesByName 15
 - getStackTrace 261
 - getStartTime 38
 - getStepSizeFraction 55
 - getStoichiometry 29
 - getStopped 34
 - getSubMenu 112
 - getSubstanceUnitsString 86
 - getSymbol 74
 - getSymbol 225
 - getSymbolEvaluationPostProcessor 14
 - getSymbolEvaluationPostProcessor 220
 - getSymbolName 23
 - getSymbols 14
 - getTime 73
 - getTimeLastNonzeroValue 139
 - getTimeOfLastUpdateMillis 36
 - getTimePoint 140
 - getTimePoints 140
 - getTimeSeriesOutputFormatAliasesList 121
 - getUnindexedValue 74
 - getUnindexedValue 221
 - getUnindexedValue 222
 - getValue 86
 - getValue 139
 - getValue 208
 - getValue 209
 - getValue 211
 - getValue 221
 - getValue 222
 - getValue 225
 - getValue 227
 - getValueArray 218
 - getValueAt 132
 - getValueAt 144
 - getValueAt 180
 - getValueAt 183
 - getValueAt 194
 - getValueAtNoFormatting 182
 - getValues 140
 - getValueWithCaching 227
 - getVersion 91
- ## H
- HalfLorentz 234
 - HalfNormal 236
 - handleCancel 152
 - handleCommandLine 121
 - handleCommandLine 257
 - handleCommandLineError 256
 - handleCopy 105
 - handleCut 105
 - handleDetailedButton 145
 - handleOutputFileAlreadyExists 147
 - handlePaste 105
 - handlePauseOrCancel 34
 - handleResize 105
 - handleResize 123
 - handleView 107
 - hasArrayIndex 217
 - hasDelayedReactionSolvers 44
 - hasLocalSymbols 25
 - hasNonzeroValue 139
 - hasValue 85
 - hasValue 74
 - hasValue 221
 - hasValue 222
-

-
- HEIGHT 115
 - HELP 109
 - HELP_ABOUT 111
 - HELP_ARG 256
 - HELP_BROWSER 111
 - HelpBrowser 149

 - I**

 - IconFactory 150
 - ImageTransferHandler 151
 - IncludeHandler 264
 - IndexedPriorityQueue 129
 - initialize 11
 - initialize 97
 - initialize 49
 - initialize 61
 - initialize 63
 - initialize 66
 - initialize 139
 - initialize 159
 - initialize 164
 - initialize 170
 - initialize 190
 - initializeDynamicSymbolAdjustmentVectors 45
 - initializePoissonEventGenerator 58
 - initializeRandomNumberGenerator 58
 - initializeSimulator 45
 - initializeSimulatorStochastic 59
 - initializeTauLeap 66
 - initializeTauLeap 68
 - initializeTauLeap 69
 - insert 135
 - insertPoint 140
 - insertRoot 135
 - integerValue 211
 - INTERNAL_SYMBOL_PREFIX 13
 - InvalidInputException 265
 - isCellEditable 182
 - isExpression 227
 - isFunctionName 201
 - isInitialized 11
 - isInitialized 44
 - isReservedSymbol 32
 - isReservedSymbol 212
 - isSimpleNumber 202
 - isStochasticSimulator 97
 - isStochasticSimulator 45
 - isStochasticSimulator 50
 - isStochasticSimulator 59
 - isValidSymbol 17
 - isValidSymbol 200
 - isWithinIncludeFile 264

 - ITEM 92
 - iterate 48
 - iterate 51
 - iterate 53
 - iterate 59
 - iterate 61
 - iterate 63
 - iterate 66

 - K**

 - KEY_DELIMITER 184

 - L**

 - Laplace 239
 - LEVEL1_VERSION1 91
 - LEVEL1_VERSION2 91
 - LINEAR 176
 - ListQueue 131
 - LN10 206
 - loadFileToEditBuffer 106
 - loadFromFile 180
 - loadFromFile 194
 - log10 207
 - LOGARITHM 161
 - Logistic 241
 - Lorentz 243

 - M**

 - mAbstractComparator 134
 - mAffectedElements 174
 - main 107
 - main 121
 - main 80
 - main 133

main 143
main 153
main 159
main 163
main 164
main 170
main 190
main 203
main 214
main 216
main 253
main 255
MainApp 107
MainMenu 108
mAllowedError 65
mAlphaParameter 174
MarkupLanguageImporter 83
MathFunctions 206
MatrixString 132
MatrixStringSelectorDialog 152
max 197
MAX_NUM_SYMBOLS_TO_PLOT 123
Maxwell 245
mBestFitDistribution 187
mCode 156
mCombinedEffectiveSignificances 174
mDelayedReactionSolvers 42
mDynamicSymbolAdjustmentVectors 43
mDynamicSymbolDelayedReactionAssociations 57
mDynamicSymbolNames 42
mDynamicSymbols 42
mDynamicSymbolValues 43
mean 230
mean 232
mean 234
mean 236
mean 238
mean 239
mean 241
mean 244
mean 245
mean 248
mean 249
mergeDataArray 179
mErrorTolerance 165
mFinalError 166
mFirstChild 136
mFixNonpositiveValues 165
MIN_NUM_BINS 167
MIN_NUM_HISTORY_BINS 6
MIN_NUM_RESULTS_TIME_POINTS 11
mInitialDynamicSymbolValues 42
mInitialized 42
mIsStochasticSimulator 42
mIterationSummaryAffected 175
mIterationSummaryFalseNegatives 175
mLastIterationWasLeap 65
mMaxIterations 165
mMethod 165
mMinNumMillisecondsForUpdate 42
mMinRatioOfLeapTimeToReactionTimeScale 65
mModelName 41
mNonDynamicSymbolExpressionValues 41
mNonDynamicSymbolValues 41
mNormalizedObservations 166
mNumAffected 174
mNumIterations 166
mNumIterations 174
mNumNonLeapIterationsSinceLastLeapCheck 65
Model 13
ModelBuilderCommandLanguage 16
ModelBuilderMarkupLanguage 87
ModelExporter 113
ModelExporterCommandLanguage 18
ModelExporterException 19
ModelExporterHumanReadable 20
ModelExporterMarkupLanguage 89
ModelExporterOrrellColumnFormat 21
ModelViewerCytoscape 114
ModelViewerException 22
ModelViewerHumanReadable 115
modifyDefaultSimulatorParameters 58
modifyDefaultSimulatorParameters 62
modifyDefaultSimulatorParameters 64
modifyDefaultSimulatorParameters 66
modifyResult 93
modifyResult 219
modifySymbol 93
modifySymbol 219
MODULE_DISPLAY_NAME 80

-
- MODULE_UNIQUE_NAME 80
 - MOLE 92
 - mParent 136
 - mPayload 136
 - mPoissonEventGenerator 57
 - mRandomNumberGenerator 57
 - mReactionHasLocalSymbolsFlags 65
 - mReactionProbabilities 43
 - mReactionRates 44
 - mReactions 43
 - mReactionsDelayedReactionAssociations 44
 - mReactionsLocalParamSymbolsMaps 43
 - mReactionsProductsDynamic 43
 - mReactionsProductsSpecies 43
 - mReactionsProductsStoichiometries 43
 - mReactionsReactantsDynamic 43
 - mReactionsReactantsSpecies 43
 - mReactionsReactantsStoichiometries 43
 - mReactionSymbols 44
 - mReducedChiSquare 187
 - mRKScratchPad 48
 - mRoot 134
 - mScale 165
 - mSecondChild 136
 - mSignificanceDistributionSeparation 174
 - mSignificances 187
 - mSimulationController 42
 - mSimulationProgressReporter 42
 - mSubtreePopulation 136
 - mSymbol 224
 - mSymbolEvaluator 41
 - mSymbolMap 42
 - mSymbolMap 222
 - multiply 203
 - mUseExpressionValueCaching 41
 - MutableBoolean 208
 - MutableDouble 209
 - MutableInteger 211
 - mValue 224
 - mWeights 174

 - N**

 - name 231
 - name 233

 - name 235
 - name 237
 - name 238
 - name 240
 - name 242
 - name 244
 - name 246
 - name 248
 - name 250
 - NAMESPACE_IDENTIFIER 13
 - negate 203
 - Node 136
 - NONE 156
 - NORM_ONLY 161
 - Normal 247
 - normalize 162
 - NULL_ARRAY_INDEX 217
 - NULL_ARRAY_INDEX 220
 - NULL_MODIFIER 256
 - NULL_OBSERVATION_STRING 178
 - NULL_REACTION 41
 - NULL_SIGNIFICANCE_STRING 193

 - O**

 - ObservationsData 178
 - ObservationsTableModel 181
 - ONE 200
 - open 105
 - openReaderForIncludeFile 264

 - P**

 - Parameter 23
 - ParserPicker 116
 - pdf 230
 - pdf 232
 - pdf 234
 - pdf 236
 - pdf 238
 - pdf 239
 - pdf 241
 - pdf 243
 - pdf 245
 - pdf 247
-

- pdf 249
PDF_NONPARAMETRIC 185
PDF_PARAMETRIC 185
peekIndex 129
peekNext 131
peekNext 134
peekNext 138
placeInCascadeFormat 148
placeInCenterOfScreen 148
POWER 176
PreferencesHandler 184
prepareForSimulation 44
prepareForStochasticSimulation 59
prepareForStochasticSimulation 61
prepareForStochasticSimulation 63
prepareForStochasticSimulation 66
print 229
printDoubleVector 259
printRegistry 255
printSymbol 205
printUsage 121
printUsage 256
PriorityQueue 134
processFileName 116
processModel 106
PRODUCT 31
- Q**
- QUANTILE 160
Queue 138
- R**
- RATIO_WIDTH_TO_STDEV 243
Rayleigh 249
REACTANT 31
Reaction 25
ReactionParticipant 29
readModel 8
readModel 16
readModel 87
readModelDescription 83
readSimulationOutput 96
record 96
RegexFileFilter 154
remove 134
removeExtension 262
reportTimeSeriesSymbolValues 77
rescaleRawObservations 162
ReservedSymbolMapper 212
ReservedSymbolMapperChemCommandLanguage 32
resizeDelayedReactionSolvers 44
rk4step 49
rkqc 49
runExternalSimulation 95
runExternalSimulation 98
runExternalSimulation 100
- S**
- save 106
saveAs 106
saveEditBufferToFile 106
savePreferences 159
savePreferences 164
savePreferences 170
savePreferences 190
scalarMultiply 198
ScientificNumberFormat 213
scrubIdentifier 128
selectParserAliasFromFileName 116
selectParserAliasManually 116
SERVICE_DESCRIPTION 81
SERVICE_NAME 81
setAbsTol 215
setArray 218
setArrayIndex 217
setCancelled 34
setCellValuesEditable 181
setCombinedSignificanceCutoff 172
setComputeFluctuations 55
setController 11
setController 47
setCurrentDirectory 117
setDebug 259
setDelay 26
setDelimiter 143
setDelimiter 152
setDirectory 264

setElementName 178
setElementNamesEditable 181
setElementSortStatus 182
setEndTime 39
setEnsembleSize 56
setEvidenceName 178
setEvidenceNamesEditable 181
setEvidenceSortStatus 182
setEvidenceWeightType 172
setExpression 201
setFractionToRemove 172
setFrameLocation 149
setFrameSize 149
setFrameTitle 149
setInfinityString 213
setInitialized 45
setInitialSignificanceCutoff 171
setLocalSymbolsMap 73
setMaxAllowedAbsoluteError 56
setMaxAllowedRelativeError 56
setMaxIterations 171
setMaxReducedChiSquare 192
setMinFractionalCostChange 172
setMinimumDigitsForScientificNotation 214
setMinNumMillisecondsForUpdate 47
setMissingDataSignificance 188
setModel 117
setModelName 38
setMultipleSelectionsAllowed 152
setName 14
setNaNString 213
setNumBins 171
setNumBins 191
setNumHistoryBins 7
setNumHistoryBins 55
setNumSteps 26
setObservationsData 181
setPreference 184
setProgressReporter 11
setProgressReporter 47
setRandomNumberGenerator 58
setRate 27
setRelTol 215
setReservedSymbolMapper 13
setResultsFinalSymbolFluctuations 38
setResultsSymbolNames 39
setResultsSymbolValues 40
setResultsTimeValues 40
setSignificanceCalculationMethod 191
setSimulationFinished 36
setSimulatorAlias 39
setSimulatorParameters 39
setSingleTailed 191
setSmoothingLength 172
setSmoothingLength 191
setSpeciesPopulation 70
setStartTime 39
setStatusUpdateIntervalSeconds 12
setStatusUpdateIntervalSeconds 47
setStepSizeFraction 55
setStopped 34
setSubMenu 111
setSymbolEvaluationPostProcessor 14
setTime 73
setErrorTolerances 49
setErrorTolerances 51
setErrorTolerances 53
setupImpl 50
setupImpl 51
setupImpl 54
setValue 24
setValue 208
setValue 209
setValue 211
setValue 225
setValue 226
setValueAt 179
setValueAt 182
setVolume 3
show 103
sign 207
SignificanceCalculationResults 187
SignificanceCalculator 188
SignificanceCalculatorDriver 190
SignificanceCalculatorParams 191
SignificancesData 193
SIGNIFICANT_DIGITS_UNKNOWN 215
SignificantDigitsCalculator 215
SimpleTextArea 155
simulate 12

-
- simulate 96
 - simulate 49
 - simulate 59
 - SimulationController 34
 - SimulationDriverModule 80
 - SimulationDriverService 81
 - simulationEnding 119
 - SimulationFailedException 35
 - SimulationLauncher 117
 - simulationLauncherClosing 119
 - SimulationLauncherCommandLine 121
 - SimulationProgressReporter 36
 - SimulationResults 38
 - SimulationResultsPlot 123
 - SimulationResultsTable 124
 - simulationStarting 119
 - Simulator 44
 - SimulatorDeterministicBase 48
 - SimulatorDeterministicRungeKuttaAdaptive 51
 - SimulatorDeterministicRungeKuttaFixed 53
 - SimulatorOdeToJavaBase 95
 - SimulatorOdeToJavaRungeKuttaAdaptive 98
 - SimulatorOdeToJavaRungeKuttaImplicit 100
 - SimulatorParameters 55
 - SimulatorStochasticBase 57
 - SimulatorStochasticGibsonBruck 61
 - SimulatorStochasticGillespie 63
 - SimulatorStochasticTauLeapBase 65
 - SimulatorStochasticTauLeapComplex 68
 - SimulatorStochasticTauLeapSimple 69
 - size 135
 - SlidingWindowTimeSeriesQueue 139
 - SPACE 127
 - Species 70
 - square 202
 - stats 207
 - SteadyStateAnalyzer 72
 - subtract 197
 - subtract 203
 - SUCCESS 120
 - sumElements 197
 - Symbol 217
 - SYMBOL_AVOGADRO 32
 - SYMBOL_TIME 32
 - SymbolEvaluationPostProcessor 219
 - SymbolEvaluationPostProcessorChemMarkupLanguage 93
 - SymbolEvaluator 220
 - SymbolEvaluatorChem 73
 - SymbolEvaluatorHashMap 222
 - SymbolValue 224
- ## T
- TAB 127
 - thetaFunction 206
 - TimeSeriesSymbolValuesReporter 77
 - toFront 118
 - TOOLS 109
 - TOOLS_EXPORT 111
 - TOOLS_RELOAD 111
 - TOOLS_SIMULATE 111
 - TOOLS_VIEW 111
 - toString 109
 - toString 111
 - toString 4
 - toString 6
 - toString 15
 - toString 24
 - toString 28
 - toString 30
 - toString 31
 - toString 91
 - toString 92
 - toString 55
 - toString 71
 - toString 75
 - toString 127
 - toString 133
 - toString 135
 - toString 177
 - toString 186
 - toString 201
 - toString 208
 - toString 210
 - toString 211
 - toString 218
 - toString 227
- ## U
-

UNIFORM 176
update 7
update 129
updateDecimalFormatSymbols 75
updateProgressStatistics 36
updateSymbolValuesForReaction 58

V

Value 226
variance 231
variance 232
variance 234
variance 236
variance 238
variance 239
variance 241
variance 244
variance 245
variance 248
variance 249
viewModel 114
viewModel 115
viewModel 10
visit 202
visit 204

W

waitForUpdate 36
WIDTH 115
writeModel 8
writeModel 16
writeModel 87
writeToFile 180

Z

ZERO 200
zeroElements 197
zeroNegativeElements 197