# Action Java Compiler Runtime Reference Manual

Generated by Doxygen 1.2.8.1

Fri Sep 13 20:48:14 2002

# Contents


**1 Action Java Compiler Runtime Package List** **1**

1.1 Action Java Compiler Runtime Package List . . . . . 1

**2 Action Java Compiler Runtime Hierarchical Index** **3**

2.1 Action Java Compiler Runtime Class Hierarchy . . . . . 3

**3 Action Java Compiler Runtime Compound Index** **5**

3.1 Action Java Compiler Runtime Compound List . . . . . 5

**4 Action Java Compiler Runtime File Index** **7**

4.1 Action Java Compiler Runtime File List . . . . . 7

**5 Action Java Compiler Runtime Package Documentation** **9**

5.1 Package ajc . . . . . 9

5.2 Package ajc.data . . . . . 10

5.3 Package ajc.data.pure . . . . . 11

**6 Action Java Compiler Runtime Class Documentation** **13**

6.1 AbstractEnactable Class Reference . . . . . 13

6.2 AbstractInfixCombinator Class Reference . . . . . 15

6.3 AbstractPrefixCombinator Class Reference . . . . . 16

6.4 Action Interface Reference . . . . . 18

6.5 ActionImpl Class Reference . . . . . 20

6.6 Agent Interface Reference . . . . . 24

6.7 AgentImpl Class Reference . . . . . 25

6.8 And Class Reference . . . . . 26

6.9 AndExceptionally Class Reference . . . . . 28

6.10 AndThen Class Reference . . . . . 30

6.11 Bindable Interface Reference . . . . . 32

6.12 Bindings Interface Reference . . . . . 33

6.13 BindingsImpl Class Reference . . . . . 35
6.14 Bool Interface Reference . . . . . 37
6.15 BoolImpl Class Reference . . . . . 38
6.16 Cell Interface Reference . . . . . 40
6.17 CellImpl Class Reference . . . . . 41
6.18 Comparable Class Reference . . . . . 42
6.19 Data Interface Reference . . . . . 43
6.20 DataConst Interface Reference . . . . . 45
6.21 DataConstImpl Class Reference . . . . . 46
6.22 DataFactory Interface Reference . . . . . 47
6.23 DataImpl Class Reference . . . . . 49
6.24 DataValue Interface Reference . . . . . 52
6.25 DataValueImpl Class Reference . . . . . 53
6.26 Datum Interface Reference . . . . . 55
6.27 DatumImpl Class Reference . . . . . 56
6.28 Empty Interface Reference . . . . . 57
6.29 EmptyImpl Class Reference . . . . . 58
6.30 Enactable Interface Reference . . . . . 60
6.31 Exception Class Reference . . . . . 61
6.32 Exceptional Class Reference . . . . . 62
6.33 Exceptionally Class Reference . . . . . 63
6.34 Failed Class Reference . . . . . 65
6.35 Hashtable Class Reference . . . . . 66
6.36 Hence Class Reference . . . . . 67
6.37 Indivisibly Class Reference . . . . . 69
6.38 Int Interface Reference . . . . . 71
6.39 IntImpl Class Reference . . . . . 73
6.40 List Interface Reference . . . . . 76
6.41 ListImpl Class Reference . . . . . 78
6.42 Message Interface Reference . . . . . 80
6.43 MessageTag Interface Reference . . . . . 81
6.44 MessageTagImpl Class Reference . . . . . 82
6.45 Otherwise Class Reference . . . . . 83
6.46 Provide Class Reference . . . . . 85
6.47 PureFactory Class Reference . . . . . 86
6.48 Runnable Class Reference . . . . . 89

6.49 Schedule Class Reference . . . 90
6.50 Scheduling Interface Reference . . . 92
6.51 Storable Interface Reference . . . 94
6.52 Store Class Reference . . . 95
6.53 Storing Interface Reference . . . 97
6.54 TaggedBuffers Class Reference . . . 98
6.55 Text Interface Reference . . . 99
6.56 TextImpl Class Reference . . . 100
6.57 Then Class Reference . . . 102
6.58 Token Interface Reference . . . 104
6.59 TokenImpl Class Reference . . . 105
6.60 Unfold Class Reference . . . 107

**7 Action Java Compiler Runtime File Documentation 109**

7.1 AbstractAction.java File Reference . . . 109
7.2 AbstractEnactable.java File Reference . . . 110
7.3 AbstractInfixCombinator.java File Reference . . . 111
7.4 AbstractPrefixCombinator.java File Reference . . . 112
7.5 Action.java File Reference . . . 113
7.6 ActionImpl.java File Reference . . . 114
7.7 Agent.java File Reference . . . 115
7.8 AgentImpl.java File Reference . . . 116
7.9 Bindable.java File Reference . . . 117
7.10 Bindings.java File Reference . . . 118
7.11 BindingsImpl.java File Reference . . . 119
7.12 Bool.java File Reference . . . 120
7.13 BoolImpl.java File Reference . . . 121
7.14 Cell.java File Reference . . . 122
7.15 CellImpl.java File Reference . . . 123
7.16 Data.java File Reference . . . 124
7.17 DataConst.java File Reference . . . 125
7.18 DataConstImpl.java File Reference . . . 126
7.19 DataFactory.java File Reference . . . 127
7.20 DataImpl.java File Reference . . . 128
7.21 DataValue.java File Reference . . . 129
7.22 DataValueImpl.java File Reference . . . 130
7.23 Datum.java File Reference . . . 131

7.24 DatumImpl.java File Reference . . . . . 132
7.25 Empty.java File Reference . . . . . 133
7.26 EmptyImpl.java File Reference . . . . . 134
7.27 Enactable.java File Reference . . . . . 135
7.28 Exceptional.java File Reference . . . . . 136
7.29 Failed.java File Reference . . . . . 137
7.30 Int.java File Reference . . . . . 138
7.31 IntImpl.java File Reference . . . . . 139
7.32 List.java File Reference . . . . . 140
7.33 ListImpl.java File Reference . . . . . 141
7.34 Message.java File Reference . . . . . 142
7.35 MessageTag.java File Reference . . . . . 143
7.36 MessageTagImpl.java File Reference . . . . . 144
7.37 PureFactory.java File Reference . . . . . 145
7.38 Schedule.java File Reference . . . . . 146
7.39 Scheduling.java File Reference . . . . . 147
7.40 Storable.java File Reference . . . . . 148
7.41 Store.java File Reference . . . . . 149
7.42 Storing.java File Reference . . . . . 150
7.43 TaggedBuffers.java File Reference . . . . . 151
7.44 Text.java File Reference . . . . . 152
7.45 TextImpl.java File Reference . . . . . 153
7.46 Token.java File Reference . . . . . 154
7.47 TokenImpl.java File Reference . . . . . 155
7.48 Tuple.java File Reference . . . . . 156
7.49 Unfold.java File Reference . . . . . 157

# Chapter 1

# Action Java Compiler Runtime Package List

## 1.1 Action Java Compiler Runtime Package List

Here are the packages with brief descriptions (if available):

**ajc** . . . . . 9
**ajc.data** . . . . . 10
**ajc.data.pure** . . . . . 11

# Chapter 2

# Action Java Compiler Runtime Hierarchical Index

## 2.1 Action Java Compiler Runtime Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:


- Bindable . . . 32
  - Datum . . . 55
    - Action . . . 18
      - ActionImpl . . . 20
        - AbstractInfixCombinator . . . 15
          - And . . . 26
          - AndExceptionally . . . 28
          - AndThen . . . 30
          - Exceptionally . . . 63
          - Hence . . . 67
          - Otherwise . . . 83
          - Then . . . 102
        - AbstractPrefixCombinator . . . 16
          - Indivisibly . . . 69
        - Provide . . . 85
    - Agent . . . 24
      - AgentImpl . . . 25
    - Bindings . . . 33
      - BindingsImpl . . . 35
    - Bool . . . 37
      - BoolImpl . . . 38
    - Cell . . . 40
      - CellImpl . . . 41
    - DatumImpl . . . 56
      - ActionImpl . . . 20
      - AgentImpl . . . 25
      - BindingsImpl . . . 35
      - BoolImpl . . . 38
      - CellImpl . . . 41
      - IntImpl . . . 73

ListImpl . . . 78
TextImpl . . . 100
MessageTagImpl . . . 82
TokenImpl . . . 105
Int . . . 71
IntImpl . . . 73
List . . . 76
ListImpl . . . 78
Text . . . 99
MessageTag . . . 81
MessageTagImpl . . . 82
TextImpl . . . 100
Token . . . 104
TokenImpl . . . 105
Comparable . . . 42
Token . . . 104
Data . . . 43
DataImpl . . . 49
DataValueImpl . . . 53
DataConstImpl . . . 46
EmptyImpl . . . 58
DatumImpl . . . 56
DataValue . . . 52
DataConst . . . 45
DataConstImpl . . . 46
Empty . . . 57
EmptyImpl . . . 58
DataValueImpl . . . 53
Datum . . . 55
DataFactory . . . 47
PureFactory . . . 86
Enactable . . . 60
AbstractEnactable . . . 13
Exception . . . 61
Exceptional . . . 62
Failed . . . 65
Unfold . . . 107
Hashtable . . . 66
TaggedBuffers . . . 98
Message . . . 80
Datum . . . 55
Runnable . . . 89
Action . . . 18
Scheduling . . . 92
Schedule . . . 90
Storable . . . 94
Datum . . . 55
Storing . . . 97
Store . . . 95

# Chapter 3

# Action Java Compiler Runtime Compound Index

## 3.1 Action Java Compiler Runtime Compound List

Here are the classes, structs, unions and interfaces with brief descriptions:


**AbstractEnactable** . . . . . 13
**AbstractInfixCombinator** . . . . . 15
**AbstractPrefixCombinator** . . . . . 16
**Action** . . . . . 18
**ActionImpl** . . . . . 20
**Agent** . . . . . 24
**AgentImpl** . . . . . 25
**And** . . . . . 26
**AndExceptionally** . . . . . 28
**AndThen** . . . . . 30
**Bindable** . . . . . 32
**Bindings** . . . . . 33
**BindingsImpl** . . . . . 35
**Bool** . . . . . 37
**BoolImpl** . . . . . 38
**Cell** . . . . . 40
**CellImpl** . . . . . 41
**Comparable** . . . . . 42
**Data** . . . . . 43
**DataConst** . . . . . 45
**DataConstImpl** . . . . . 46
**DataFactory** . . . . . 47
**DataImpl** . . . . . 49
**DataValue** . . . . . 52
**DataValueImpl** . . . . . 53
**Datum** . . . . . 55
**DatumImpl** . . . . . 56
**Empty** . . . . . 57
**EmptyImpl** . . . . . 58
**Enactable** . . . . . 60

**Exception** . . . 61
**Exceptional** . . . 62
**Exceptionally** . . . 63
**Failed** . . . 65
**Hashtable** . . . 66
**Hence** . . . 67
**Indivisibly** . . . 69
**Int** . . . 71
**IntImpl** . . . 73
**List** . . . 76
**ListImpl** . . . 78
**Message** . . . 80
**MessageTag** . . . 81
**MessageTagImpl** . . . 82
**Otherwise** . . . 83
**Provide** . . . 85
**PureFactory** . . . 86
**Runnable** . . . 89
**Schedule** . . . 90
**Scheduling** . . . 92
**Storable** . . . 94
**Store** . . . 95
**Storing** . . . 97
**TaggedBuffers** . . . 98
**Text** . . . 99
**TextImpl** . . . 100
**Then** . . . 102
**Token** . . . 104
**TokenImpl** . . . 105
**Unfold** (Describe class `Unfold` here) . . . 107

# Chapter 4

# Action Java Compiler Runtime File Index

## 4.1 Action Java Compiler Runtime File List

Here is a list of all files with brief descriptions:


**AbstractAction.java** . . . 109
**AbstractEnactable.java** . . . 110
**AbstractInfixCombinator.java** . . . 111
**AbstractPrefixCombinator.java** . . . 112
**Action.java** . . . 113
**ActionImpl.java** . . . 114
**Agent.java** . . . 115
**AgentImpl.java** . . . 116
**Bindable.java** . . . 117
**Bindings.java** . . . 118
**BindingsImpl.java** . . . 119
**Bool.java** . . . 120
**BoolImpl.java** . . . 121
**Cell.java** . . . 122
**CellImpl.java** . . . 123
**Data.java** . . . 124
**DataConst.java** . . . 125
**DataConstImpl.java** . . . 126
**DataFactory.java** . . . 127
**DataImpl.java** . . . 128
**DataValue.java** . . . 129
**DataValueImpl.java** . . . 130
**Datum.java** . . . 131
**DatumImpl.java** . . . 132
**Empty.java** . . . 133
**EmptyImpl.java** . . . 134
**Enactable.java** . . . 135
**Exceptional.java** . . . 136
**Failed.java** . . . 137
**Int.java** . . . 138

**IntImpl.java** . . . 139
**List.java** . . . 140
**ListImpl.java** . . . 141
**Message.java** . . . 142
**MessageTag.java** . . . 143
**MessageTagImpl.java** . . . 144
**PureFactory.java** . . . 145
**Schedule.java** . . . 146
**Scheduling.java** . . . 147
**Storable.java** . . . 148
**Store.java** . . . 149
**Storing.java** . . . 150
**TaggedBuffers.java** . . . 151
**Text.java** . . . 152
**TextImpl.java** . . . 153
**Token.java** . . . 154
**TokenImpl.java** . . . 155
**Tuple.java** . . . 156
**Unfold.java** . . . 157

# Chapter 5

# Action Java Compiler Runtime Package Documentation

## 5.1 Package ajc

### Interfaces

- interface Enactable
- interface Scheduling
- interface Storing

### Classes

- class AbstractEnactable
- class Exceptional
- class Failed
- class Schedule
- class Store
- class TaggedBuffers
- class Unfold

  *Describe class* `Unfold` *here.*

## 5.2 Package ajc.data

### Interfaces

- interface Action
- interface Agent
- interface Bindable
- interface Bindings
- interface Bool
- interface Cell
- interface Data
- interface DataConst
- interface DataFactory
- interface DataValue
- interface Datum
- interface Empty
- interface Int
- interface List
- interface Message
- interface MessageTag
- interface Storable
- interface Text
- interface Token

## 5.3 Package ajc.data.pure

### Classes

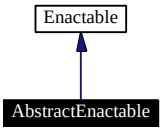
- class **And::_And**
- class **AndExceptionally::_AndExceptionally**
- class **AndThen::_AndThen**
- class **Exceptionally::_Exceptionally**
- class **Hence::_Hence**
- class **Indivisibly::_Indivisibly**
- class **Otherwise::_Otherwise**
- class **Provide::_Provide**
- class **Then::_Then**
- class AbstractInfixCombinator
- class AbstractPrefixCombinator
- class ActionImpl
- class AgentImpl
- class And
- class AndExceptionally
- class AndThen
- class BindingsImpl
- class BoolImpl
- class CellImpl
- class DataConstImpl
- class DataImpl
- class DataValueImpl
- class DatumImpl
- class EmptyImpl
- class Exceptionally
- class Hence
- class Indivisibly
- class IntImpl
- class ListImpl
- class MessageTagImpl
- class Otherwise
- class Provide
- class PureFactory
- class TextImpl
- class Then
- class TokenImpl

# Chapter 6

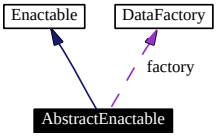
# Action Java Compiler Runtime Class Documentation

## 6.1 AbstractEnactable Class Reference

Inheritance diagram for AbstractEnactable:




Collaboration diagram for AbstractEnactable:




### Protected Methods

- Data fail () throws Failed
- Data raise (Data data) throws Exceptional
- Data catchExceptional (Exception e) throws Failed

### Protected Attributes

- DataFactory factory

### 6.1.1 Member Function Documentation

#### 6.1.1.1 Data AbstractEnactable::catchExceptional (Exception *e*) [protected]

Definition at line 25 of file AbstractEnactable.java.

#### 6.1.1.2 Data AbstractEnactable::fail () [protected]

Definition at line 14 of file AbstractEnactable.java.

#### 6.1.1.3 Data AbstractEnactable::raise (Data *data*) [protected]

Definition at line 19 of file AbstractEnactable.java.

### 6.1.2 Member Data Documentation

#### 6.1.2.1 DataFactory AbstractEnactable::factory [protected]

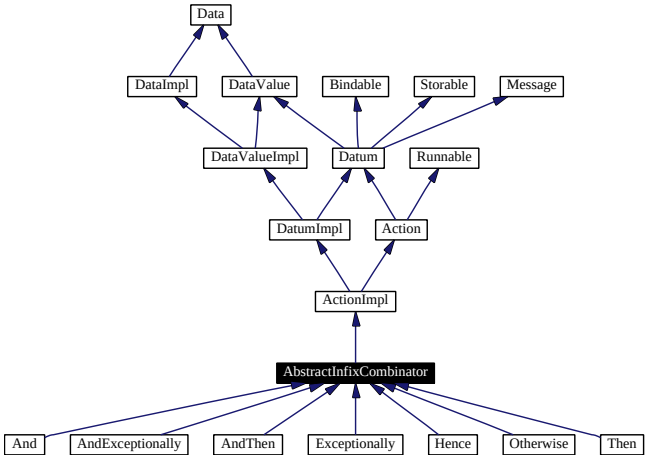
Definition at line 12 of file AbstractEnactable.java.

The documentation for this class was generated from the following file:

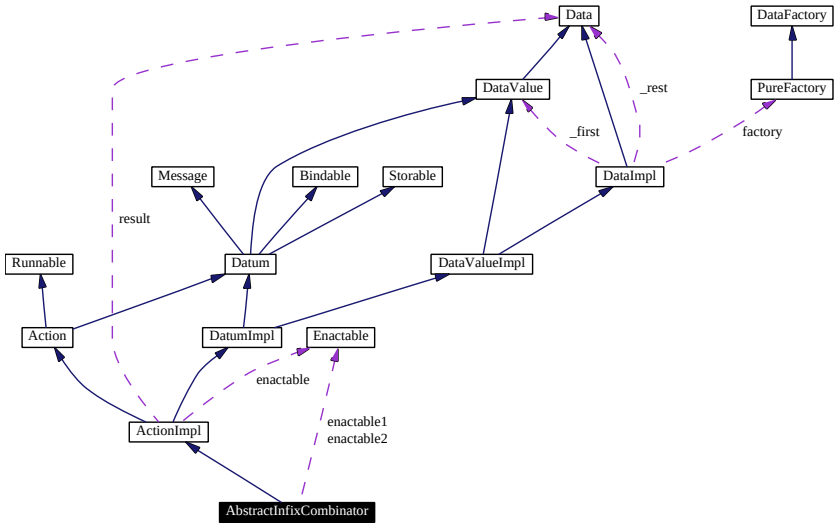
- AbstractEnactable.java

## 6.2 AbstractInfixCombinator Class Reference

Inheritance diagram for AbstractInfixCombinator:




Collaboration diagram for AbstractInfixCombinator:

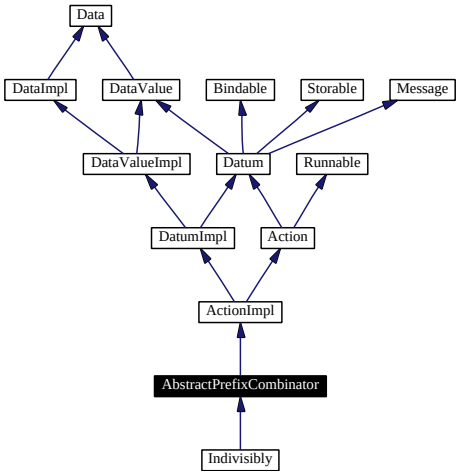



The documentation for this class was generated from the following file:

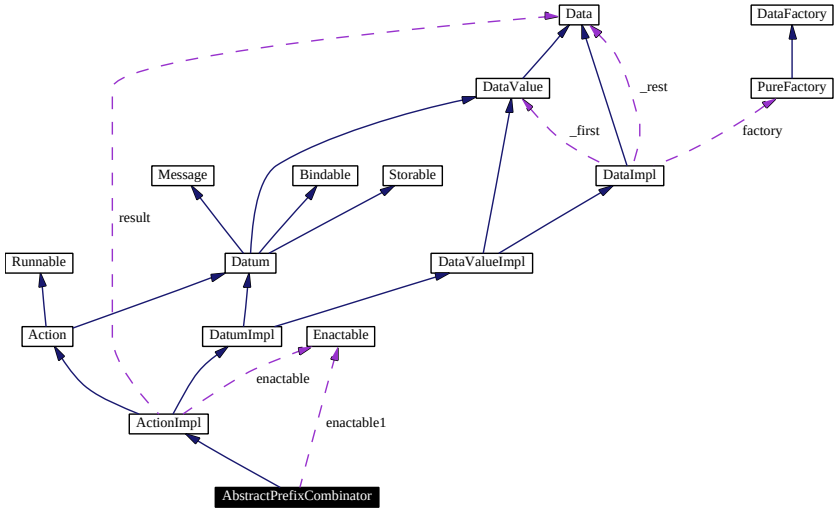
- AbstractInfixCombinator.java

## 6.3 AbstractPrefixCombinator Class Reference

Inheritance diagram for AbstractPrefixCombinator:




Collaboration diagram for AbstractPrefixCombinator:




### Protected Attributes

- Enactable enactable1

### 6.3.1 Member Data Documentation

#### 6.3.1.1 Enactable AbstractPrefixCombinator::enactable1 [protected]

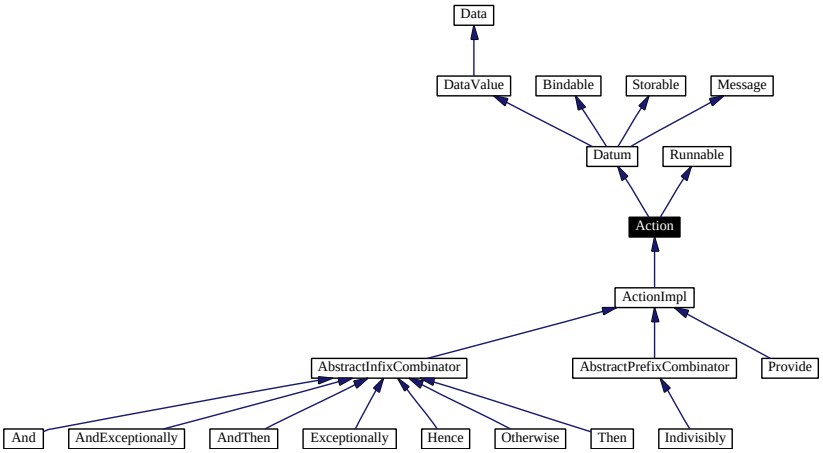
Definition at line 8 of file AbstractPrefixCombinator.java.

The documentation for this class was generated from the following file:

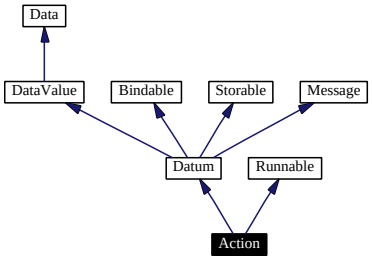
- AbstractPrefixCombinator.java

## 6.4 Action Interface Reference

Inheritance diagram for Action:




Collaboration diagram for Action:




### Public Methods

- Enactable enactableValue ()
- Data enact () throws Exceptional, Failed, Unfold
- Action then (Action action)
- Action andThen (Action action)
- Action and (Action action)
- Action exceptionally (Action action)
- Action andExceptionally (Action action)
- Action otherwise (Action action)
- Action hence (Action action)
- Action indivisibly ()

### 6.4.1 Member Function Documentation

#### 6.4.1.1 Action Action::and (Action *action*)

Reimplemented in ActionImpl.

#### 6.4.1.2 Action Action::andExceptionally (Action *action*)

Reimplemented in ActionImpl.

#### 6.4.1.3 Action Action::andThen (Action *action*)

Reimplemented in ActionImpl.

#### 6.4.1.4 Data Action::enact ()

Reimplemented in ActionImpl, and ActionImpl.

Referenced by ActionImpl::run().

#### 6.4.1.5 Enactable Action::enactableValue ()

Reimplemented in ActionImpl.

Referenced by ActionImpl::and(), ActionImpl::andExceptionally(), ActionImpl::andThen(), ActionImpl::exceptionally(), ActionImpl::hence(), ActionImpl::otherwise(), and ActionImpl::then().

#### 6.4.1.6 Action Action::exceptionally (Action *action*)

Reimplemented in ActionImpl.

#### 6.4.1.7 Action Action::hence (Action *action*)

Reimplemented in ActionImpl.

#### 6.4.1.8 Action Action::indivisibly ()

Reimplemented in ActionImpl.

#### 6.4.1.9 Action Action::otherwise (Action *action*)

Reimplemented in ActionImpl.

#### 6.4.1.10 Action Action::then (Action *action*)

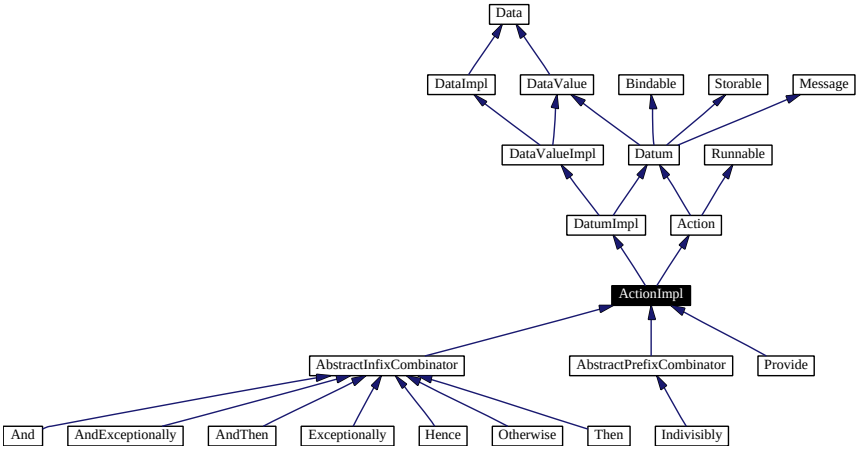
Reimplemented in ActionImpl.

The documentation for this interface was generated from the following file:

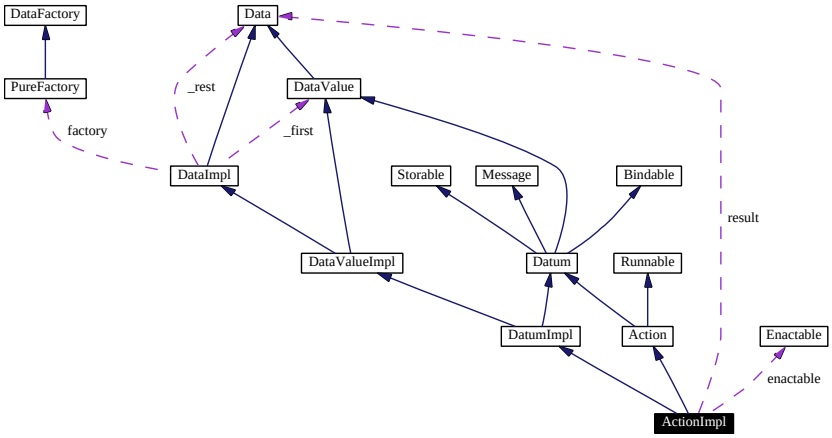
- Action.java

## 6.5 ActionImpl Class Reference

Inheritance diagram for ActionImpl:




Collaboration diagram for ActionImpl:




### Public Methods

- Enactable getEnactable ()
- Data enact ()
- Action giveThen (Action action)
- Action giveAndThen (Action action)
- Action giveAnd (Action action)
- Action giveExceptionally (Action action)
- Action giveAndExceptionally (Action action)
- Action giveOtherwise (Action action)
- Action giveHence (Action action)
- Action giveIndivisibly ()
- Enactable enactableValue ()
- Data enact () throws Exceptional, Failed, Unfold

- Action then (Action action)
- Action andThen (Action action)
- Action and (Action action)
- Action exceptionally (Action action)
- Action andExceptionally (Action action)
- Action otherwise (Action action)
- Action hence (Action action)
- Action indivisibly ()
- void run ()
- Empty equals (Data data) throws Exceptional

## 6.5.1 Member Function Documentation

### 6.5.1.1 Action ActionImpl::and (Action *action*)

Reimplemented from Action.

Definition at line 38 of file ActionImpl.java.

### 6.5.1.2 Action ActionImpl::andExceptionally (Action *action*)

Reimplemented from Action.

Definition at line 44 of file ActionImpl.java.

### 6.5.1.3 Action ActionImpl::andThen (Action *action*)

Reimplemented from Action.

Definition at line 35 of file ActionImpl.java.

### 6.5.1.4 Data ActionImpl::enact ()

Reimplemented from Action.

Definition at line 28 of file ActionImpl.java.

### 6.5.1.5 Data ActionImpl::enact ()

Reimplemented from Action.

Definition at line 32 of file AbstractAction.java.

### 6.5.1.6 Enactable ActionImpl::enactableValue ()

Reimplemented from Action.

Definition at line 24 of file ActionImpl.java.

#### 6.5.1.7 Empty ActionImpl::equals (Data *data*)

Reimplemented from DataValueImpl.

Definition at line 68 of file ActionImpl.java.

#### 6.5.1.8 Action ActionImpl::exceptionally (Action *action*)

Reimplemented from Action.

Definition at line 41 of file ActionImpl.java.

#### 6.5.1.9 Enactable ActionImpl::getEnactable ()

Definition at line 28 of file AbstractAction.java.

#### 6.5.1.10 Action ActionImpl::giveAnd (Action *action*)

Definition at line 42 of file AbstractAction.java.

#### 6.5.1.11 Action ActionImpl::giveAndExceptionally (Action *action*)

Definition at line 48 of file AbstractAction.java.

#### 6.5.1.12 Action ActionImpl::giveAndThen (Action *action*)

Definition at line 39 of file AbstractAction.java.

#### 6.5.1.13 Action ActionImpl::giveExceptionally (Action *action*)

Definition at line 45 of file AbstractAction.java.

#### 6.5.1.14 Action ActionImpl::giveHence (Action *action*)

Definition at line 54 of file AbstractAction.java.

#### 6.5.1.15 Action ActionImpl::giveIndivisibly ()

Definition at line 57 of file AbstractAction.java.

#### 6.5.1.16 Action ActionImpl::giveOtherwise (Action *action*)

Definition at line 51 of file AbstractAction.java.

#### 6.5.1.17 Action ActionImpl::giveThen (Action *action*)

Definition at line 36 of file AbstractAction.java.

#### 6.5.1.18 Action ActionImpl::hence (Action *action*)

Reimplemented from Action.

Definition at line 50 of file ActionImpl.java.

#### 6.5.1.19 Action ActionImpl::indivisibly ()

Reimplemented from Action.

Definition at line 53 of file ActionImpl.java.

#### 6.5.1.20 Action ActionImpl::otherwise (Action *action*)

Reimplemented from Action.

Definition at line 47 of file ActionImpl.java.

#### 6.5.1.21 void ActionImpl::run ()

Definition at line 58 of file ActionImpl.java.

#### 6.5.1.22 Action ActionImpl::then (Action *action*)

Reimplemented from Action.

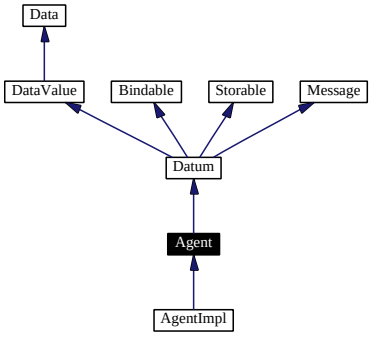
Definition at line 32 of file ActionImpl.java.

The documentation for this class was generated from the following files:

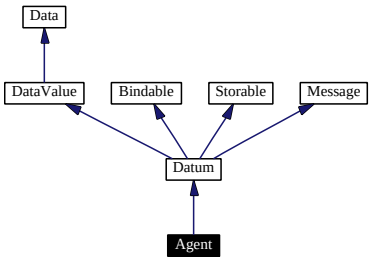
- AbstractAction.java
- ActionImpl.java

## 6.6 Agent Interface Reference

Inheritance diagram for Agent:




Collaboration diagram for Agent:

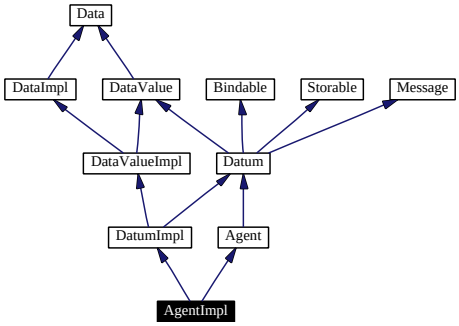



The documentation for this interface was generated from the following file:

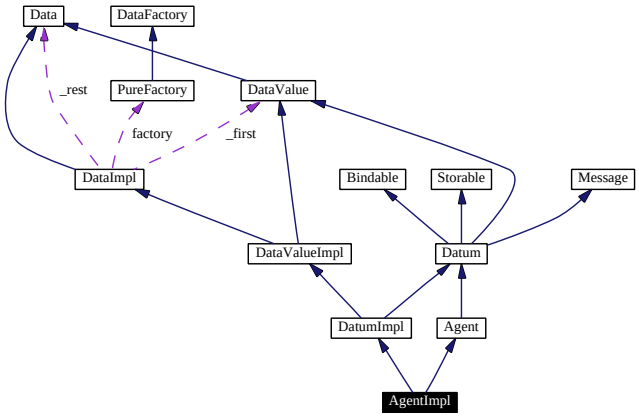
- Agent.java

# 6.7 AgentImpl Class Reference

Inheritance diagram for AgentImpl:




Collaboration diagram for AgentImpl:

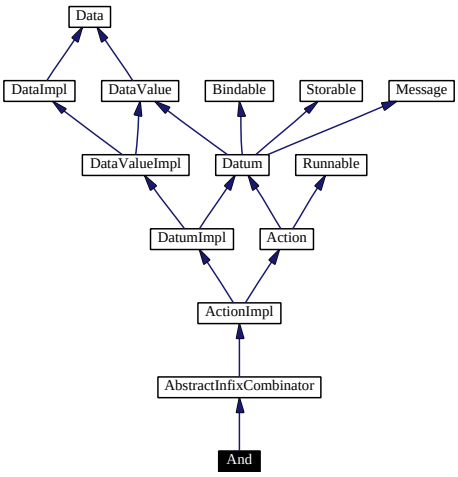



The documentation for this class was generated from the following file:

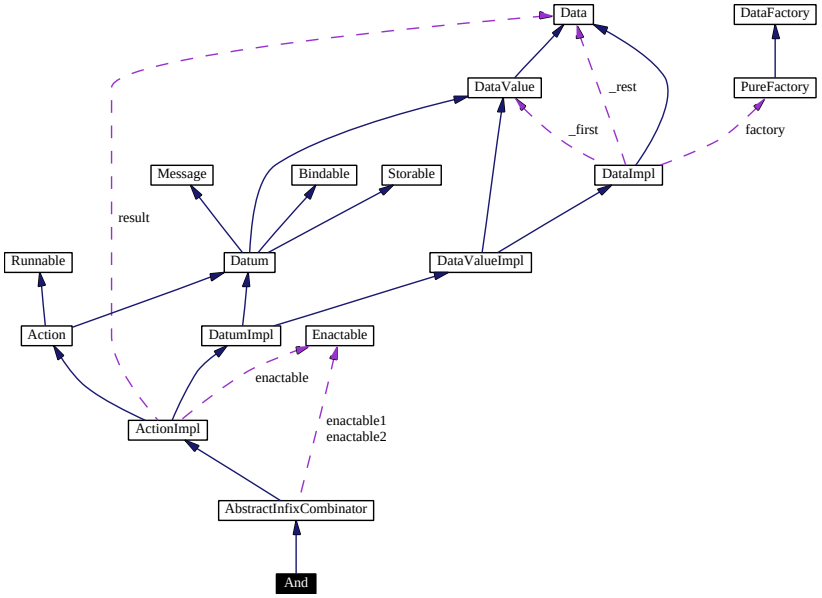
- AgentImpl.java

## 6.8 And Class Reference

Inheritance diagram for And:




Collaboration diagram for And:




### Public Methods

- And (PureFactory factory, Enactable enactable1, Enactable enactable2)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- And (PureFactory factory, Enactable enactable1, Enactable enactable2)

## 6.8.1 Constructor & Destructor Documentation

### 6.8.1.1 And::And (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 84 of file AbstractAction.java.

### 6.8.1.2 And::And (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 112 of file ActionImpl.java.

## 6.8.2 Member Function Documentation

### 6.8.2.1 Data And::enact (Data *data*, Bindings *bindings*)

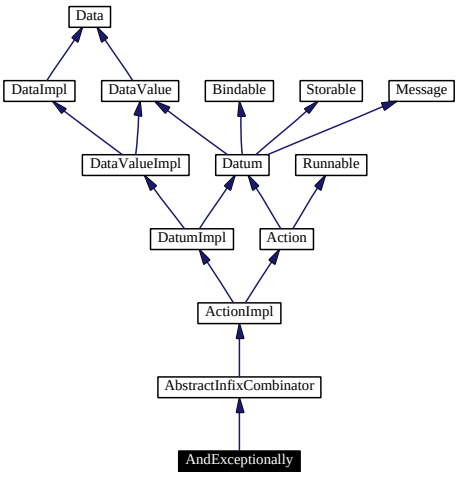
Definition at line 87 of file AbstractAction.java.

The documentation for this class was generated from the following files:

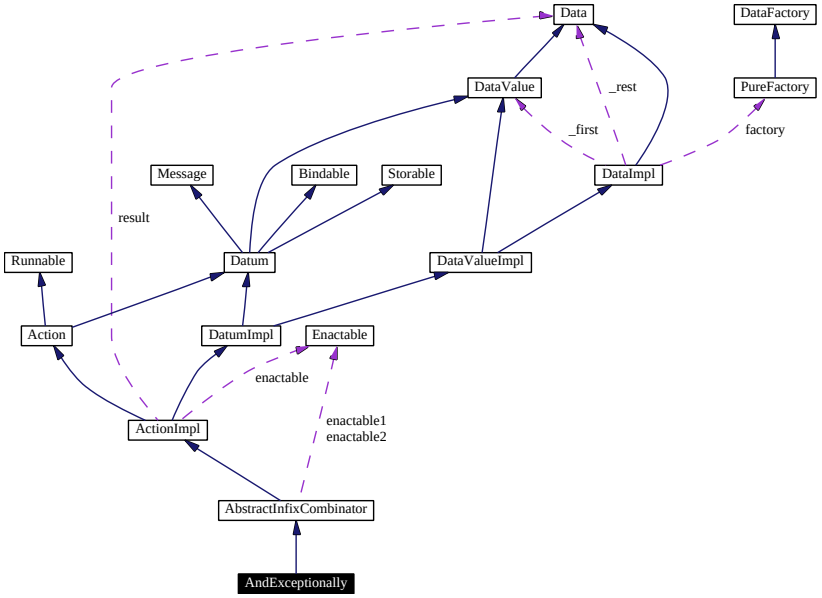
- AbstractAction.java
- ActionImpl.java

## 6.9 AndExceptionally Class Reference

Inheritance diagram for AndExceptionally:




Collaboration diagram for AndExceptionally:




### Public Methods

- AndExceptionally (PureFactory factory, Enactable enactable1, Enactable enactable2)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- AndExceptionally (PureFactory factory, Enactable enactable1, Enactable enactable2)

## 6.9.1 Constructor & Destructor Documentation

### 6.9.1.1 AndExceptionally::AndExceptionally (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 110 of file AbstractAction.java.

### 6.9.1.2 AndExceptionally::AndExceptionally (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 154 of file ActionImpl.java.

## 6.9.2 Member Function Documentation

### 6.9.2.1 Data AndExceptionally::enact (Data *data*, Bindings *bindings*)

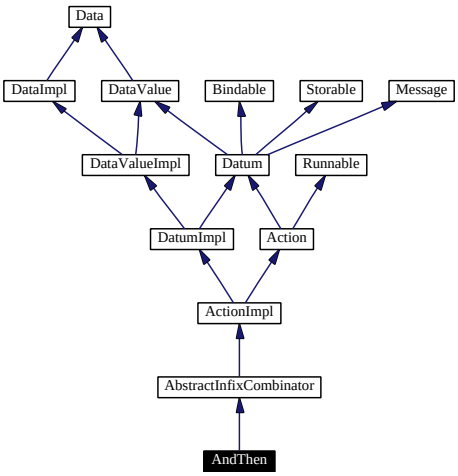
Definition at line 113 of file AbstractAction.java.

The documentation for this class was generated from the following files:

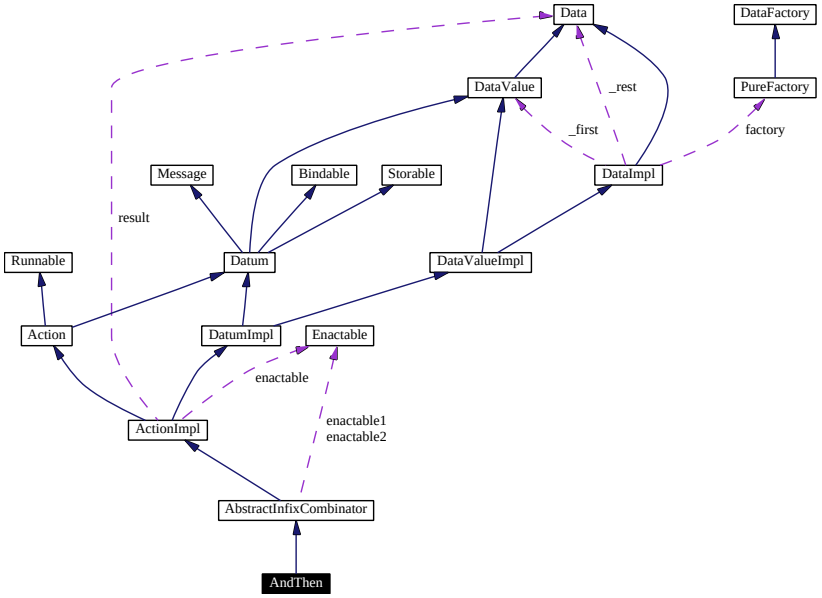
- AbstractAction.java
- ActionImpl.java

## 6.10 AndThen Class Reference

Inheritance diagram for AndThen:




Collaboration diagram for AndThen:




### Public Methods

- AndThen (PureFactory factory, Enactable enactable1, Enactable enactable2)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- AndThen (PureFactory factory, Enactable enactable1, Enactable enactable2)

## 6.10.1 Constructor & Destructor Documentation

### 6.10.1.1 AndThen::AndThen (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 73 of file AbstractAction.java.

### 6.10.1.2 AndThen::AndThen (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 98 of file ActionImpl.java.

## 6.10.2 Member Function Documentation

### 6.10.2.1 Data AndThen::enact (Data *data*, Bindings *bindings*)

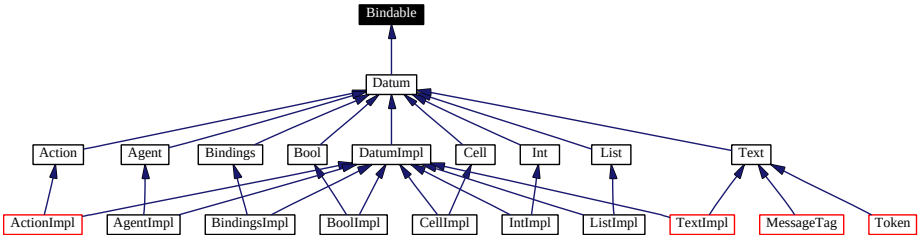
Definition at line 76 of file AbstractAction.java.

The documentation for this class was generated from the following files:

- AbstractAction.java
- ActionImpl.java

## 6.11 Bindable Interface Reference

Inheritance diagram for Bindable:

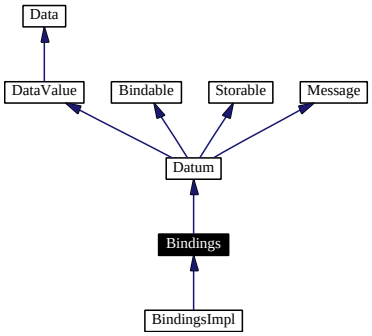



The documentation for this interface was generated from the following file:

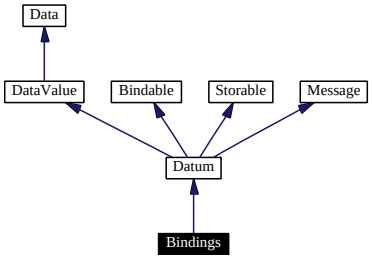
- Bindable.java

## 6.12 Bindings Interface Reference

Inheritance diagram for Bindings:




Collaboration diagram for Bindings:




### Public Methods

- Map mapValue ()
- Bindable bound (Token token) throws Exceptional
- Bindings overriding (Bindings bindings)
- Bindings disjointUnion (Bindings bindings)

### 6.12.1 Member Function Documentation

#### 6.12.1.1 Bindable Bindings::bound (Token *token*)

Reimplemented in BindingsImpl.

#### 6.12.1.2 Bindings Bindings::disjointUnion (Bindings *bindings*)

Reimplemented in BindingsImpl.

#### 6.12.1.3 Map Bindings::mapValue ()

Reimplemented in BindingsImpl.

Referenced by BindingsImpl::disjointUnion(), and BindingsImpl::overriding().

#### 6.12.1.4 Bindings Bindings::overriding (Bindings *bindings*)

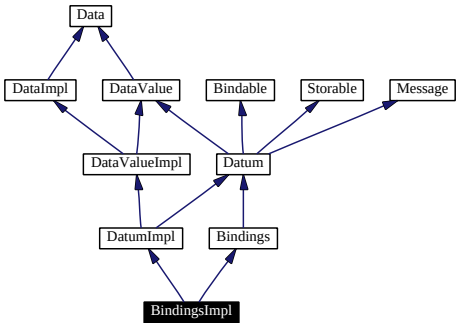
Reimplemented in BindingsImpl.

The documentation for this interface was generated from the following file:

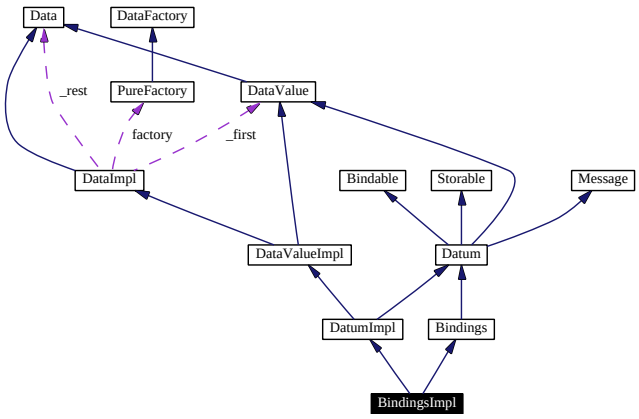
- Bindings.java

## 6.13 BindingsImpl Class Reference

Inheritance diagram for BindingsImpl:




Collaboration diagram for BindingsImpl:




### Public Methods

- Map mapValue ()
- Bindable bound (Token token) throws Exceptional
- Bindings overriding (Bindings bindings)
- Bindings disjointUnion (Bindings bindings)
- String toString ()

### 6.13.1 Member Function Documentation

#### 6.13.1.1 Bindable BindingsImpl::bound (Token *token*)

Reimplemented from Bindings.

Definition at line 39 of file BindingsImpl.java.

#### 6.13.1.2 Bindings BindingsImpl::disjointUnion (Bindings *bindings*)

Reimplemented from Bindings.

Definition at line 52 of file BindingsImpl.java.

#### 6.13.1.3 Map BindingsImpl::mapValue ()

Reimplemented from Bindings.

Definition at line 29 of file BindingsImpl.java.

#### 6.13.1.4 Bindings BindingsImpl::overriding (Bindings *bindings*)

Reimplemented from Bindings.

Definition at line 46 of file BindingsImpl.java.

#### 6.13.1.5 String BindingsImpl::toString ()

Reimplemented from DataImpl.

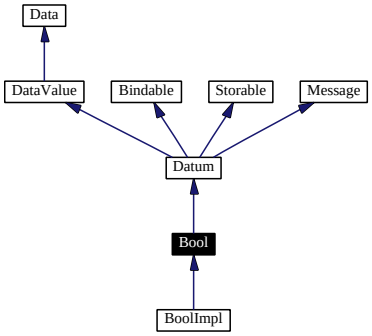
Definition at line 65 of file BindingsImpl.java.

The documentation for this class was generated from the following file:

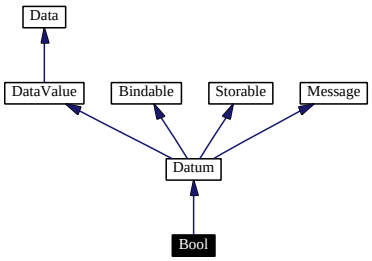
- BindingsImpl.java

# 6.14 Bool Interface Reference

Inheritance diagram for Bool:




Collaboration diagram for Bool:




## Public Methods

- boolean booleanValue ()
- Bool not ()

## 6.14.1 Member Function Documentation

### 6.14.1.1 boolean Bool::booleanValue ()

Reimplemented in BoolImpl.

### 6.14.1.2 Bool Bool::not ()

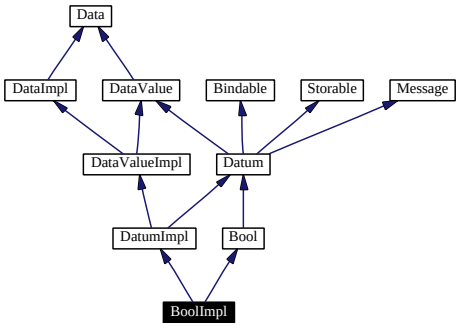
Reimplemented in BoolImpl.

The documentation for this interface was generated from the following file:

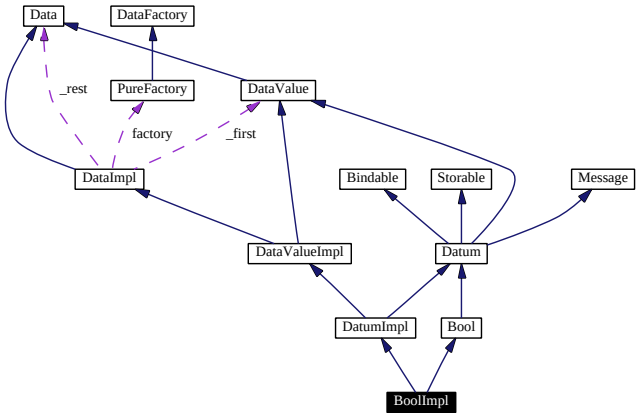
- Bool.java

## 6.15 BoolImpl Class Reference

Inheritance diagram for BoolImpl:




Collaboration diagram for BoolImpl:




### Public Methods

- boolean booleanValue ()
- Bool not ()
- Empty equals (Data data) throws Exceptional

### 6.15.1 Member Function Documentation

#### 6.15.1.1 boolean BoolImpl::booleanValue ()

Reimplemented from Bool.

Definition at line 17 of file BoolImpl.java.

#### 6.15.1.2 Empty BoolImpl::equals (Data *data*)

Reimplemented from DataValueImpl.

Definition at line 25 of file BoolImpl.java.

#### 6.15.1.3 Bool BoolImpl::not ()

Reimplemented from Bool.

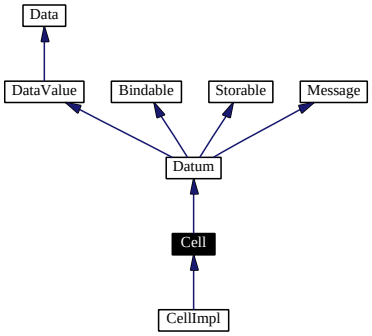
Definition at line 21 of file BoolImpl.java.

The documentation for this class was generated from the following file:

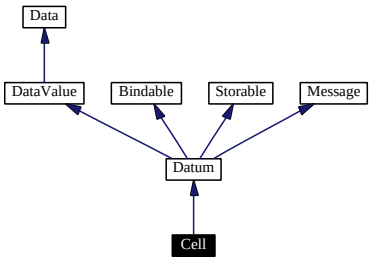
- BoolImpl.java

## 6.16 Cell Interface Reference

Inheritance diagram for Cell:




Collaboration diagram for Cell:

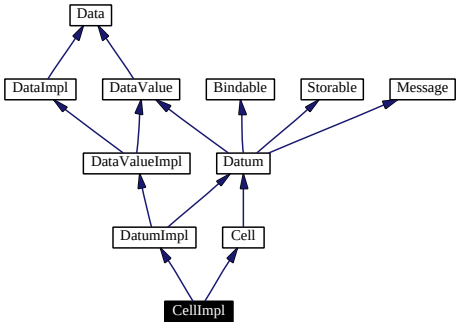



The documentation for this interface was generated from the following file:

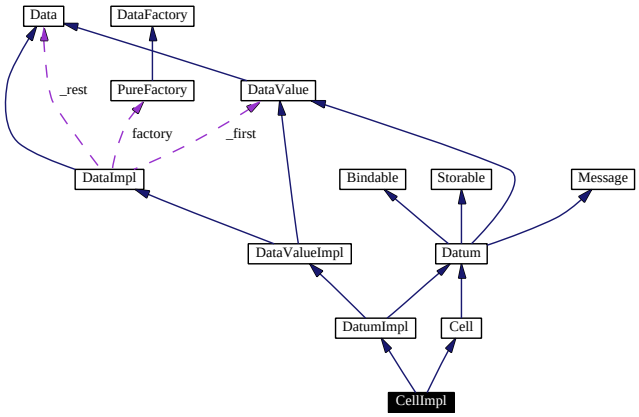
- Cell.java

# 6.17 CellImpl Class Reference

Inheritance diagram for CellImpl:




Collaboration diagram for CellImpl:




## Public Methods

- String toString ()

## 6.17.1 Member Function Documentation

### 6.17.1.1 String CellImpl::toString ()

Reimplemented from DataImpl.

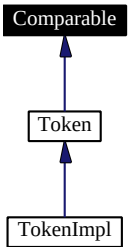
Definition at line 16 of file CellImpl.java.

The documentation for this class was generated from the following file:

- CellImpl.java

## 6.18 Comparable Class Reference

Inheritance diagram for Comparable:

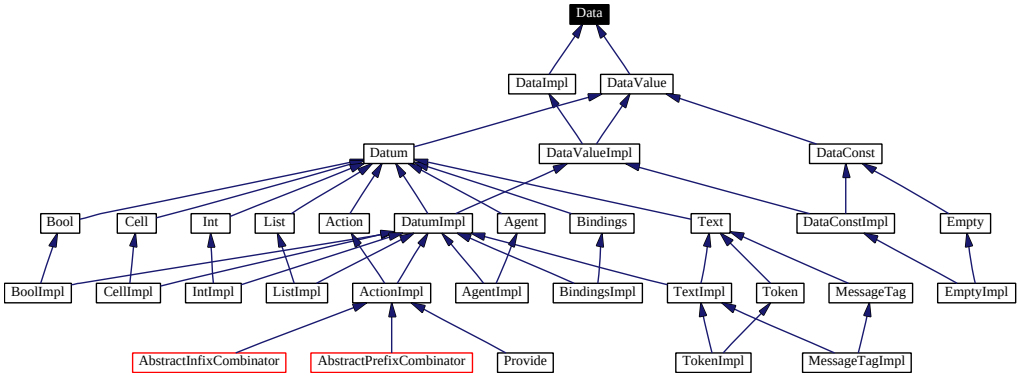



The documentation for this class was generated from the following file:

- Token.java

# 6.19 Data Interface Reference

Inheritance diagram for Data:




## Public Methods

- Data concat (Data data)
- List tupleToList ()
- DataValue first () throws Exceptional
- Data rest ()
- DataValue at (int n) throws Exceptional
- Action provide ()

## 6.19.1 Member Function Documentation

### 6.19.1.1 DataValue Data::at (int *n*)

Reimplemented in DataImpl, DataValueImpl, and EmptyImpl.

Referenced by DataImpl::at().

### 6.19.1.2 Data Data::concat (Data *data*)

Reimplemented in DataImpl, DataValueImpl, and EmptyImpl.

Referenced by DataImpl::concat().

### 6.19.1.3 DataValue Data::first ()

Reimplemented in DataImpl, DataValueImpl, and EmptyImpl.

### 6.19.1.4 Action Data::provide ()

Reimplemented in DataImpl.

#### 6.19.1.5 Data Data::rest ()

Reimplemented in DataImpl, and DataValueImpl.

#### 6.19.1.6 List Data::tupleToList ()

Reimplemented in DataImpl, DataValueImpl, and EmptyImpl.

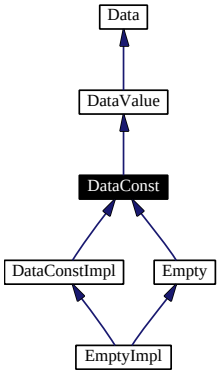
Referenced by DataImpl::tupleToList().

The documentation for this interface was generated from the following file:

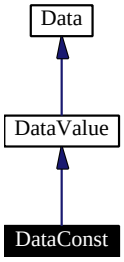
- Data.java

## 6.20 DataConst Interface Reference

Inheritance diagram for DataConst:




Collaboration diagram for DataConst:

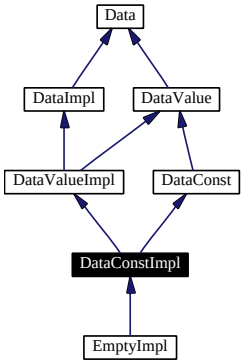



The documentation for this interface was generated from the following file:

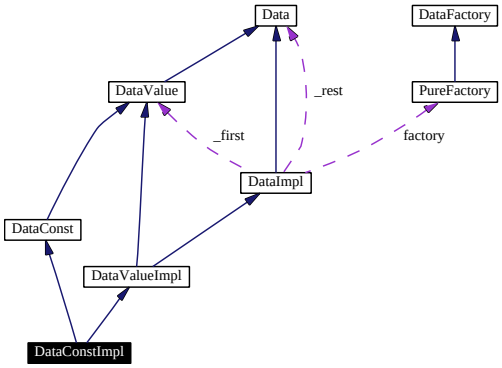
- DataConst.java

## 6.21 DataConstImpl Class Reference

Inheritance diagram for DataConstImpl:




Collaboration diagram for DataConstImpl:




### Protected Methods

- DataConstImpl (PureFactory factory)

### 6.21.1 Constructor & Destructor Documentation

#### 6.21.1.1 DataConstImpl::DataConstImpl (PureFactory *factory*) [protected]

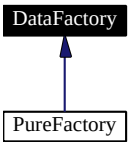
Definition at line 9 of file DataConstImpl.java.

The documentation for this class was generated from the following file:

- DataConstImpl.java

# 6.22 DataFactory Interface Reference

Inheritance diagram for DataFactory:




## Public Methods

- Empty makeEmpty ()
- Data makeTuple (DataValue dataValue, Data data)
- Int makeInt (long n)
- Token makeToken (String s)
- Bool makeBool (boolean b)
- MessageTag makeMessageTag (String s)
- List makeList1 (DataValue dataValue)
- List makeList0 ()
- Action makeAction (Enactable enactable)
- Bindings makeNoBindings ()
- Bindings makeBindings (Token token, Bindable bindable)
- Agent makeAgent ()
- Cell makeCell ()

## 6.22.1 Member Function Documentation

### 6.22.1.1 Action DataFactory::makeAction (Enactable *enactable*)

Reimplemented in PureFactory.

### 6.22.1.2 Agent DataFactory::makeAgent ()

Reimplemented in PureFactory.

Referenced by Schedule::activate().

### 6.22.1.3 Bindings DataFactory::makeBindings (Token *token*, Bindable *bindable*)

Reimplemented in PureFactory.

### 6.22.1.4 Bool DataFactory::makeBool (boolean *b*)

Reimplemented in PureFactory.

#### 6.22.1.5 Cell DataFactory::makeCell ()

Reimplemented in PureFactory.

Referenced by Store::create().

#### 6.22.1.6 Empty DataFactory::makeEmpty ()

Reimplemented in PureFactory.

Referenced by Schedule::deactivate(), TaggedBuffers::dequeue(), Store::destroy(), Store::inspect(), Schedule::receive(), Schedule::send(), and Store::update().

#### 6.22.1.7 Int DataFactory::makeInt (long *n*)

Reimplemented in PureFactory.

Referenced by Schedule::chooseNatural(), and Schedule::currentTime().

#### 6.22.1.8 List DataFactory::makeList0 ()

Reimplemented in PureFactory.

#### 6.22.1.9 List DataFactory::makeList1 (DataValue *dataValue*)

Reimplemented in PureFactory.

#### 6.22.1.10 MessageTag DataFactory::makeMessageTag (String *s*)

Reimplemented in PureFactory.

#### 6.22.1.11 Bindings DataFactory::makeNoBindings ()

Reimplemented in PureFactory.

#### 6.22.1.12 Token DataFactory::makeToken (String *s*)

Reimplemented in PureFactory.

#### 6.22.1.13 Data DataFactory::makeTuple (DataValue *dataValue*, Data *data*)

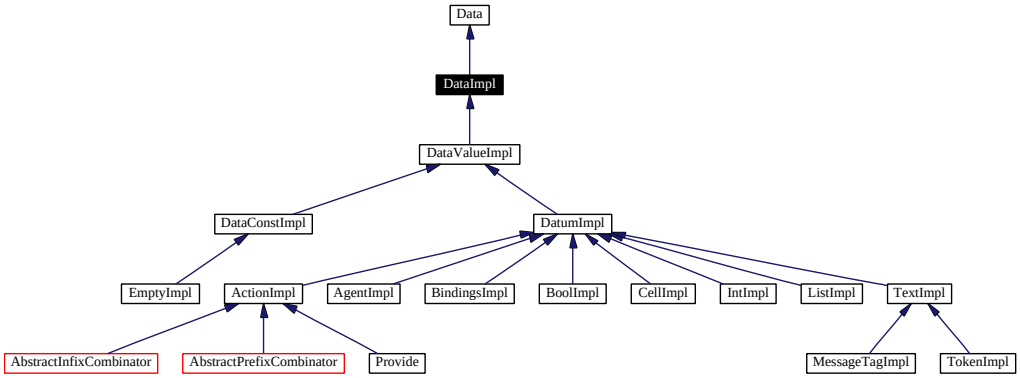
Reimplemented in PureFactory.

The documentation for this interface was generated from the following file:

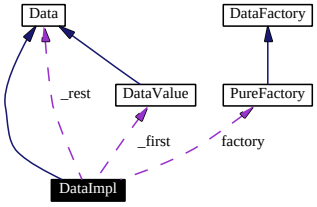
- DataFactory.java

# 6.23 DataImpl Class Reference

Inheritance diagram for DataImpl:




Collaboration diagram for DataImpl:




## Public Methods

- Data concat (Data data)
- List tupleToList ()
- DataValue first () throws Exceptional
- Data rest ()
- DataValue at (int n) throws Exceptional
- Action provide ()
- String toString ()

## Protected Methods

- DataImpl (PureFactory factory, DataValue first, Data rest)
- DataImpl (PureFactory factory)

## Protected Attributes

- DataValue _first
- Data _rest

### 6.23.1 Constructor & Destructor Documentation

#### 6.23.1.1 DataImpl::DataImpl (PureFactory *factory*, DataValue *first*, Data *rest*) [protected]

Definition at line 12 of file DataImpl.java.

#### 6.23.1.2 DataImpl::DataImpl (PureFactory *factory*) [protected]

Definition at line 18 of file DataImpl.java.

### 6.23.2 Member Function Documentation

#### 6.23.2.1 DataValue DataImpl::at (int *n*)

Reimplemented from Data.

Reimplemented in DataValueImpl, and EmptyImpl.

Definition at line 40 of file DataImpl.java.

#### 6.23.2.2 Data DataImpl::concat (Data *data*)

Reimplemented from Data.

Reimplemented in DataValueImpl, and EmptyImpl.

Definition at line 24 of file DataImpl.java.

#### 6.23.2.3 DataValue DataImpl::first ()

Reimplemented from Data.

Reimplemented in DataValueImpl, and EmptyImpl.

Definition at line 32 of file DataImpl.java.

#### 6.23.2.4 Action DataImpl::provide ()

Reimplemented from Data.

Definition at line 46 of file DataImpl.java.

#### 6.23.2.5 Data DataImpl::rest ()

Reimplemented from Data.

Reimplemented in DataValueImpl.

Definition at line 36 of file DataImpl.java.

#### 6.23.2.6 String DataImpl::toString ()

Reimplemented in BindingsImpl, CellImpl, EmptyImpl, IntImpl, ListImpl, and TokenImpl.

Definition at line 50 of file DataImpl.java.

#### 6.23.2.7 List DataImpl::tupleToList ()

Reimplemented from Data.

Reimplemented in DataValueImpl, and EmptyImpl.

Definition at line 28 of file DataImpl.java.

### 6.23.3 Member Data Documentation

#### 6.23.3.1 DataValue DataImpl::_first [protected]

Definition at line 9 of file DataImpl.java.

#### 6.23.3.2 Data DataImpl::_rest [protected]

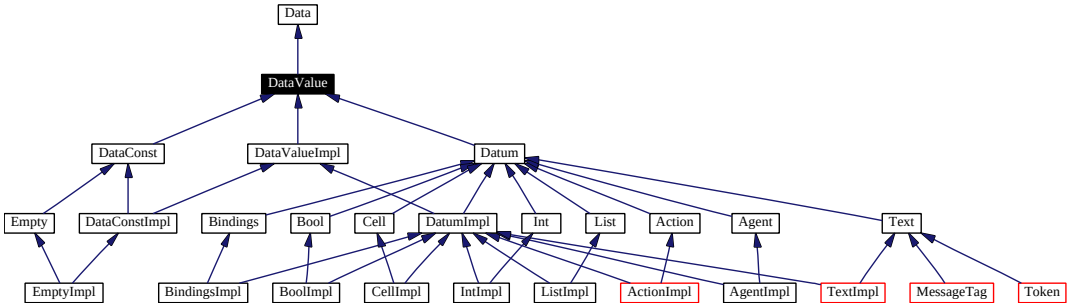
Definition at line 10 of file DataImpl.java.

The documentation for this class was generated from the following file:

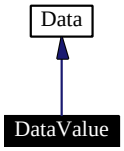
- DataImpl.java

## 6.24 DataValue Interface Reference

Inheritance diagram for DataValue:




Collaboration diagram for DataValue:




### Public Methods

- Empty equals (Data data) throws Exceptional

### 6.24.1 Member Function Documentation

#### 6.24.1.1 Empty DataValue::equals (Data *data*)

Reimplemented in ActionImpl, BoolImpl, DataValueImpl, IntImpl, ListImpl, and TextImpl.

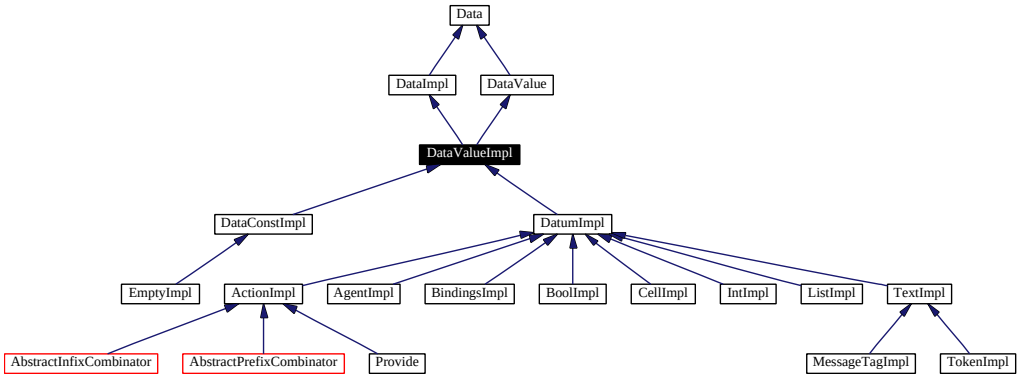
Referenced by ListImpl::equals().

The documentation for this interface was generated from the following file:

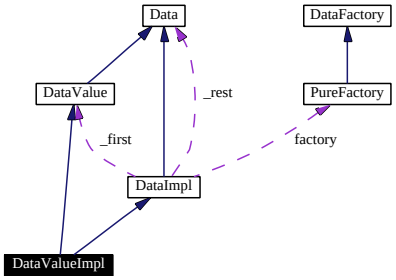
- DataValue.java

# 6.25 DataValueImpl Class Reference

Inheritance diagram for DataValueImpl:




Collaboration diagram for DataValueImpl:




## Public Methods

- DataValue first () throws Exceptional
- Data rest ()
- List tupleToList ()
- Data concat (Data data)
- DataValue at (int n) throws Exceptional
- Empty equals (Data data) throws Exceptional

## Protected Methods

- DataValueImpl (PureFactory factory)

### 6.25.1 Constructor & Destructor Documentation

#### 6.25.1.1 DataValueImpl::DataValueImpl (PureFactory *factory*) `[protected]`

Definition at line 9 of file DataValueImpl.java.

## 6.25.2 Member Function Documentation

### 6.25.2.1 DataValue DataValueImpl::at (int *n*)

Reimplemented from DataImpl.

Reimplemented in EmptyImpl.

Definition at line 27 of file DataValueImpl.java.

### 6.25.2.2 Data DataValueImpl::concat (Data *data*)

Reimplemented from DataImpl.

Reimplemented in EmptyImpl.

Definition at line 23 of file DataValueImpl.java.

### 6.25.2.3 Empty DataValueImpl::equals (Data *data*)

Reimplemented from DataValue.

Reimplemented in ActionImpl, BoolImpl, IntImpl, ListImpl, and TextImpl.

Definition at line 37 of file DataValueImpl.java.

### 6.25.2.4 DataValue DataValueImpl::first ()

Reimplemented from DataImpl.

Reimplemented in EmptyImpl.

Definition at line 12 of file DataValueImpl.java.

### 6.25.2.5 Data DataValueImpl::rest ()

Reimplemented from DataImpl.

Definition at line 15 of file DataValueImpl.java.

### 6.25.2.6 List DataValueImpl::tupleToList ()

Reimplemented from DataImpl.

Reimplemented in EmptyImpl.

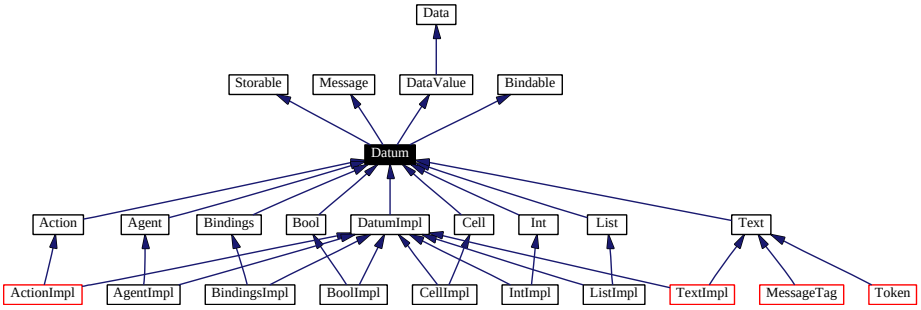
Definition at line 19 of file DataValueImpl.java.

The documentation for this class was generated from the following file:

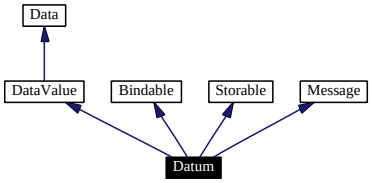
- DataValueImpl.java

# 6.26 Datum Interface Reference

Inheritance diagram for Datum:




Collaboration diagram for Datum:

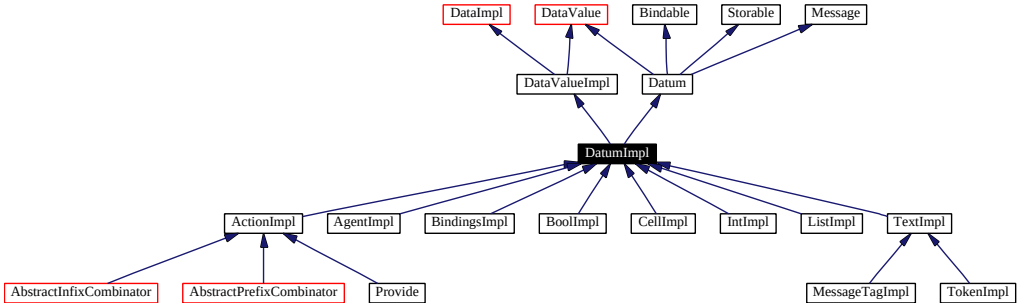



The documentation for this interface was generated from the following file:

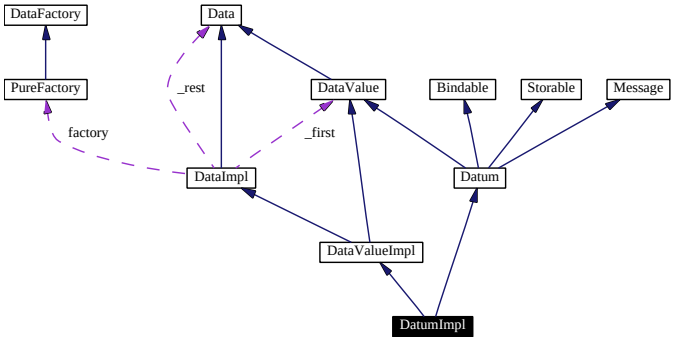
- Datum.java

# 6.27 DatumImpl Class Reference

Inheritance diagram for DatumImpl:




Collaboration diagram for DatumImpl:




## Protected Methods

- DatumImpl (PureFactory factory)

## 6.27.1 Constructor & Destructor Documentation

### 6.27.1.1 DatumImpl::DatumImpl (PureFactory *factory*) [protected]

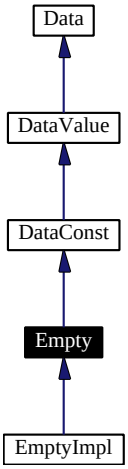
Definition at line 8 of file DatumImpl.java.

The documentation for this class was generated from the following file:

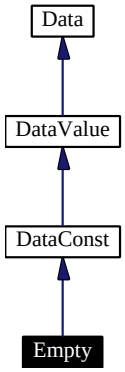
- DatumImpl.java

## 6.28 Empty Interface Reference

Inheritance diagram for Empty:




Collaboration diagram for Empty:

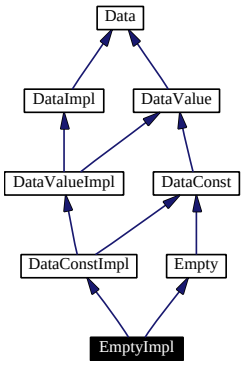



The documentation for this interface was generated from the following file:

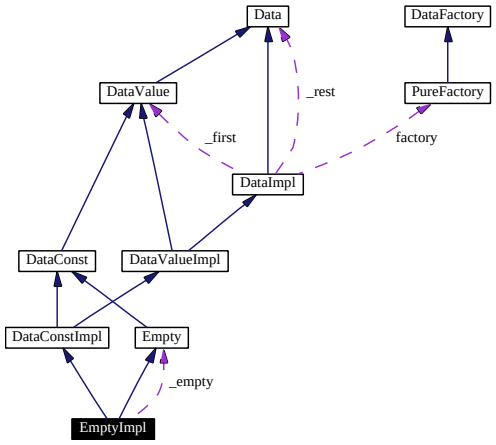
- Empty.java

## 6.29 EmptyImpl Class Reference

Inheritance diagram for EmptyImpl:




Collaboration diagram for EmptyImpl:




### Public Methods

- DataValue at (int n) throws Exceptional
- DataValue first () throws Exceptional
- Data concat (Data data)
- List tupleToList ()
- String toString ()

### 6.29.1 Member Function Documentation

#### 6.29.1.1 DataValue EmptyImpl::at (int *n*)

Reimplemented from DataValueImpl.

Definition at line 26 of file EmptyImpl.java.

#### 6.29.1.2 Data EmptyImpl::concat (Data *data*)

Reimplemented from DataValueImpl.

Definition at line 34 of file EmptyImpl.java.

#### 6.29.1.3 DataValue EmptyImpl::first ()

Reimplemented from DataValueImpl.

Definition at line 30 of file EmptyImpl.java.

#### 6.29.1.4 String EmptyImpl::toString ()

Reimplemented from DataImpl.

Definition at line 42 of file EmptyImpl.java.

#### 6.29.1.5 List EmptyImpl::tupleToList ()

Reimplemented from DataValueImpl.

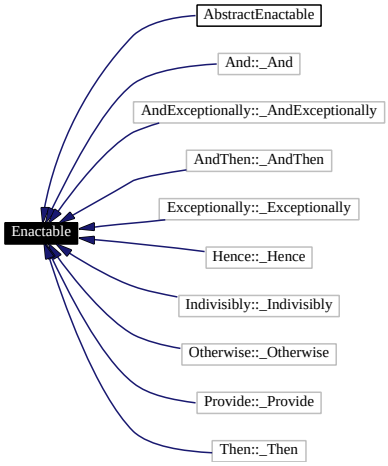
Definition at line 38 of file EmptyImpl.java.

The documentation for this class was generated from the following file:

- EmptyImpl.java

## 6.30 Enactable Interface Reference

Inheritance diagram for Enactable:




### Public Methods

- Data enact (Data data, Bindings bindings) throws Exceptional, Failed, Unfold

### 6.30.1 Member Function Documentation

#### 6.30.1.1 Data Enactable::enact (Data *data*, Bindings *bindings*)

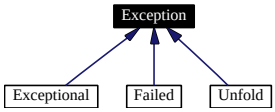
Referenced by Indivisibly::enact(), Hence::enact(), Otherwise::enact(), AndExceptionally::enact(), Exceptionally::enact(), And::enact(), AndThen::enact(), Then::enact(), and ActionImpl::enact().

The documentation for this interface was generated from the following file:

- Enactable.java

## 6.31 Exception Class Reference

Inheritance diagram for Exception:

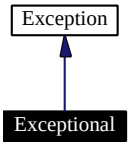



The documentation for this class was generated from the following file:

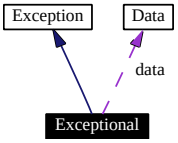
- Exceptional.java

## 6.32 Exceptional Class Reference

Inheritance diagram for Exceptional:




Collaboration diagram for Exceptional:




### Public Methods

- Data getData ()
- Exceptional (Data data)
- Exceptional (String s, Data data)

### 6.32.1 Constructor & Destructor Documentation

#### 6.32.1.1 Exceptional::Exceptional (Data *data*)

Definition at line 13 of file Exceptional.java.

#### 6.32.1.2 Exceptional::Exceptional (String *s*, Data *data*)

Definition at line 19 of file Exceptional.java.

### 6.32.2 Member Function Documentation

#### 6.32.2.1 Data Exceptional::getData ()

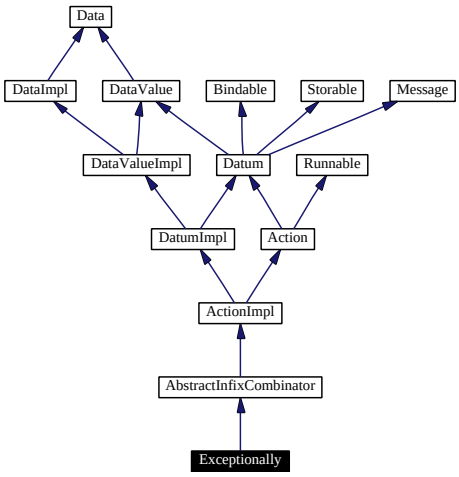
Definition at line 9 of file Exceptional.java.

The documentation for this class was generated from the following file:

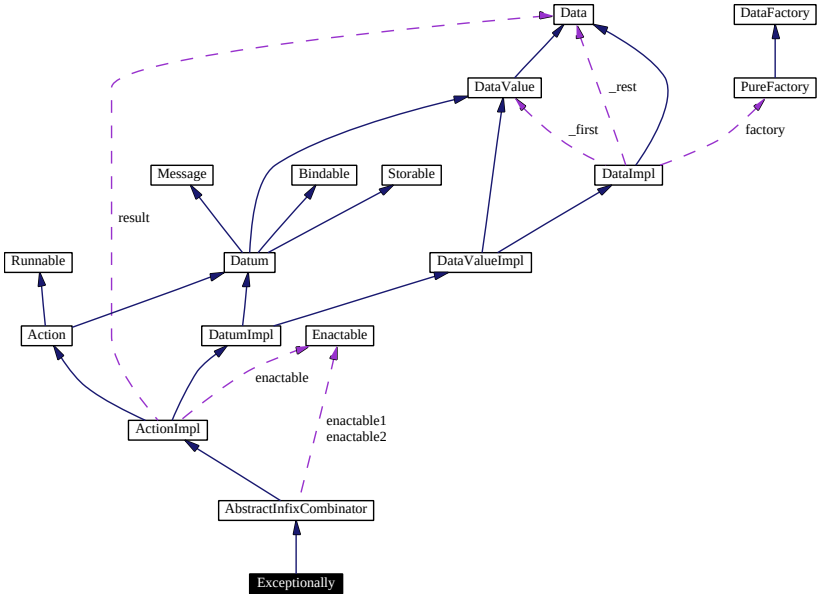
- Exceptional.java

# 6.33 Exceptionally Class Reference

Inheritance diagram for Exceptionally:




Collaboration diagram for Exceptionally:




## Public Methods

- Exceptionally (PureFactory factory, Enactable enactable1, Enactable enactable2)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- Exceptionally (PureFactory factory, Enactable enactable1, Enactable enactable2)

### 6.33.1 Constructor & Destructor Documentation

#### 6.33.1.1 Exceptionally::Exceptionally (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 95 of file AbstractAction.java.

#### 6.33.1.2 Exceptionally::Exceptionally (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 130 of file ActionImpl.java.

### 6.33.2 Member Function Documentation

#### 6.33.2.1 Data Exceptionally::enact (Data *data*, Bindings *bindings*)

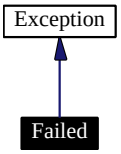
Definition at line 98 of file AbstractAction.java.

The documentation for this class was generated from the following files:

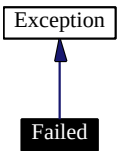
- AbstractAction.java
- ActionImpl.java

## 6.34 Failed Class Reference

Inheritance diagram for Failed:




Collaboration diagram for Failed:




### Public Methods

- Failed ()

### 6.34.1 Constructor & Destructor Documentation

#### 6.34.1.1 Failed::Failed ()

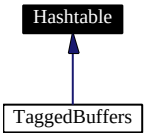
Definition at line 5 of file Failed.java.

The documentation for this class was generated from the following file:

- Failed.java

## 6.35 Hashtable Class Reference

Inheritance diagram for Hashtable:

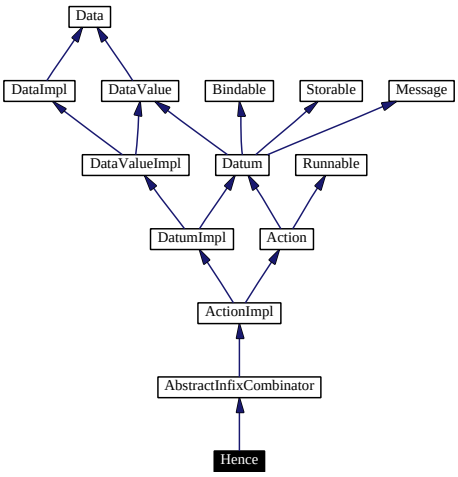



The documentation for this class was generated from the following file:

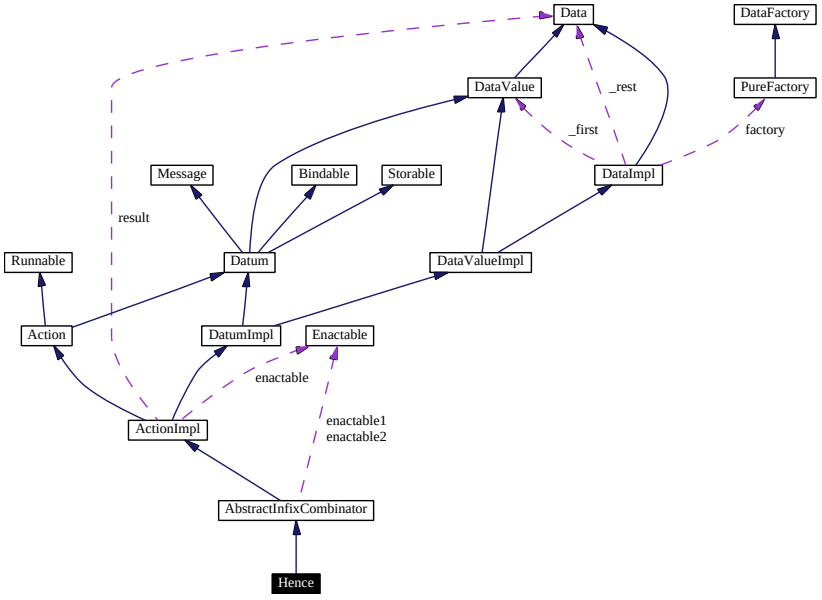
- TaggedBuffers.java

# 6.36 Hence Class Reference

Inheritance diagram for Hence:




Collaboration diagram for Hence:




## Public Methods

- Hence (PureFactory factory, Enactable enactable1, Enactable enactable2)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- Hence (PureFactory factory, Enactable enactable1, Enactable enactable2)

### 6.36.1 Constructor & Destructor Documentation

#### 6.36.1.1 Hence::Hence (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 145 of file AbstractAction.java.

#### 6.36.1.2 Hence::Hence (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 183 of file ActionImpl.java.

### 6.36.2 Member Function Documentation

#### 6.36.2.1 Data Hence::enact (Data *data*, Bindings *bindings*)

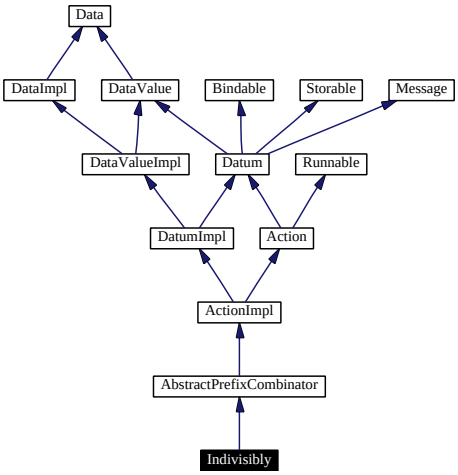
Definition at line 148 of file AbstractAction.java.

The documentation for this class was generated from the following files:

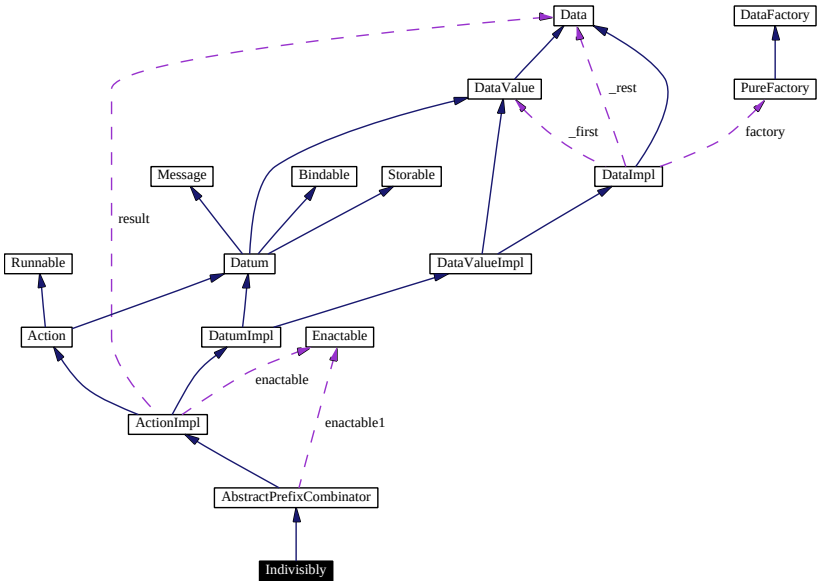
- AbstractAction.java
- ActionImpl.java

# 6.37 Indivisibly Class Reference

Inheritance diagram for Indivisibly:




Collaboration diagram for Indivisibly:




## Public Methods

- Indivisibly (PureFactory factory, Enactable action)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- Indivisibly (PureFactory factory, Enactable enactable1)

### 6.37.1 Constructor & Destructor Documentation

#### 6.37.1.1 Indivisibly::Indivisibly (PureFactory *factory*, Enactable *enactable1*)

Definition at line 156 of file AbstractAction.java.

#### 6.37.1.2 Indivisibly::Indivisibly (PureFactory *factory*, Enactable *enactable1*)

Definition at line 198 of file ActionImpl.java.

### 6.37.2 Member Function Documentation

#### 6.37.2.1 Data Indivisibly::enact (Data *data*, Bindings *bindings*)

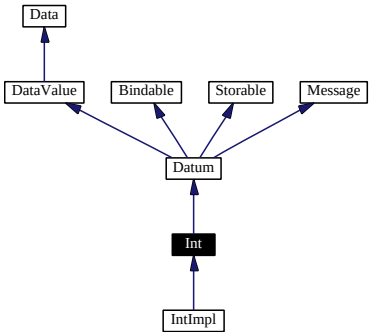
Definition at line 159 of file AbstractAction.java.

The documentation for this class was generated from the following files:

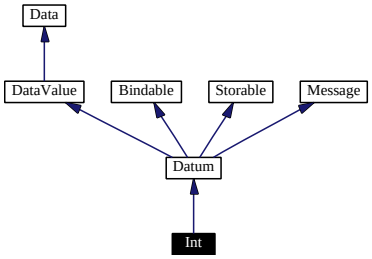
- AbstractAction.java
- ActionImpl.java

# 6.38 Int Interface Reference

Inheritance diagram for Int:




Collaboration diagram for Int:




## Public Methods

- long intValue ()
- Int plus (Int n)
- Int minus (Int n)
- Int monus (Int n)
- Int times (Int n)
- Empty greater (Int n) throws Exceptional
- Empty greaterOrEq (Int n) throws Exceptional
- Empty less (Int n) throws Exceptional
- Empty lessOrEq (Int n) throws Exceptional

## 6.38.1 Member Function Documentation

### 6.38.1.1 Empty Int::greater (Int *n*)

Reimplemented in IntImpl.

### 6.38.1.2 Empty Int::greaterOrEq (Int *n*)

Reimplemented in IntImpl.

#### 6.38.1.3 long Int::intValue ()

Reimplemented in IntImpl.

Referenced by IntImpl::greater(), IntImpl::greaterOrEq(), IntImpl::less(), IntImpl::lessOrEq(), IntImpl::minus(), IntImpl::monus(), IntImpl::plus(), and IntImpl::times().

#### 6.38.1.4 Empty Int::less (Int *n*)

Reimplemented in IntImpl.

#### 6.38.1.5 Empty Int::lessOrEq (Int *n*)

Reimplemented in IntImpl.

#### 6.38.1.6 Int Int::minus (Int *n*)

Reimplemented in IntImpl.

#### 6.38.1.7 Int Int::monus (Int *n*)

Reimplemented in IntImpl.

#### 6.38.1.8 Int Int::plus (Int *n*)

Reimplemented in IntImpl.

#### 6.38.1.9 Int Int::times (Int *n*)

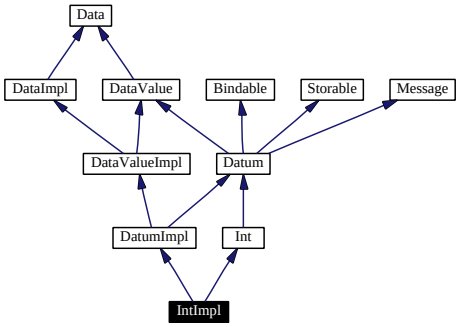
Reimplemented in IntImpl.

The documentation for this interface was generated from the following file:

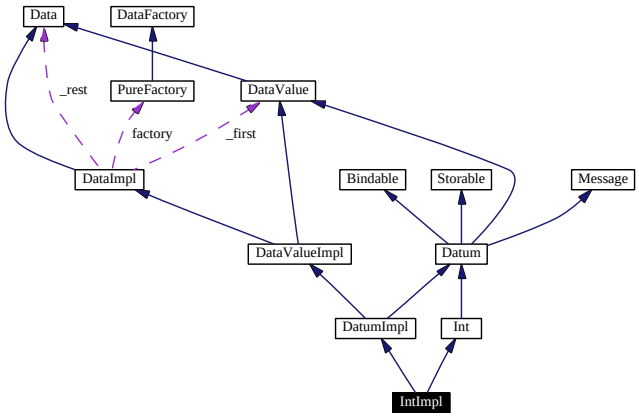
- Int.java

# 6.39 IntImpl Class Reference

Inheritance diagram for IntImpl:




Collaboration diagram for IntImpl:




## Public Methods

- long intValue ()
- Int plus (Int n)
- Int minus (Int n)
- Int monus (Int n)
- Int times (Int n)
- Empty greater (Int n) throws Exceptional
- Empty greaterOrEq (Int n) throws Exceptional
- Empty less (Int n) throws Exceptional
- Empty lessOrEq (Int n) throws Exceptional
- Empty equals (Data data) throws Exceptional
- String toString ()

## Protected Methods

- IntImpl (PureFactory factory, long value)

## 6.39.1 Constructor & Destructor Documentation

### 6.39.1.1 IntImpl::IntImpl (PureFactory *factory*, long *value*) [protected]

Definition at line 10 of file IntImpl.java.

## 6.39.2 Member Function Documentation

### 6.39.2.1 Empty IntImpl::equals (Data *data*)

Reimplemented from DataValueImpl.

Definition at line 56 of file IntImpl.java.

### 6.39.2.2 Empty IntImpl::greater (Int *n*)

Reimplemented from Int.

Definition at line 35 of file IntImpl.java.

### 6.39.2.3 Empty IntImpl::greaterOrEq (Int *n*)

Reimplemented from Int.

Definition at line 40 of file IntImpl.java.

### 6.39.2.4 long IntImpl::intValue ()

Reimplemented from Int.

Definition at line 16 of file IntImpl.java.

### 6.39.2.5 Empty IntImpl::less (Int *n*)

Reimplemented from Int.

Definition at line 45 of file IntImpl.java.

### 6.39.2.6 Empty IntImpl::lessOrEq (Int *n*)

Reimplemented from Int.

Definition at line 50 of file IntImpl.java.

### 6.39.2.7 Int IntImpl::minus (Int *n*)

Reimplemented from Int.

Definition at line 23 of file IntImpl.java.

#### 6.39.2.8 Int IntImpl::monus (Int *n*)

Reimplemented from Int.

Definition at line 26 of file IntImpl.java.

#### 6.39.2.9 Int IntImpl::plus (Int *n*)

Reimplemented from Int.

Definition at line 20 of file IntImpl.java.

#### 6.39.2.10 Int IntImpl::times (Int *n*)

Reimplemented from Int.

Definition at line 31 of file IntImpl.java.

#### 6.39.2.11 String IntImpl::toString ()

Reimplemented from DataImpl.

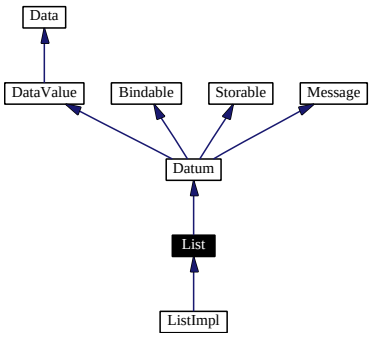
Definition at line 63 of file IntImpl.java.

The documentation for this class was generated from the following file:

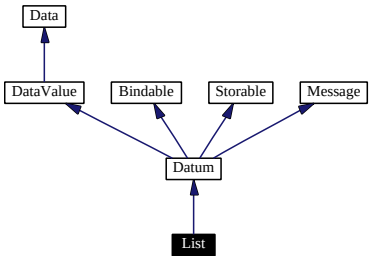
- IntImpl.java

## 6.40 List Interface Reference

Inheritance diagram for List:




Collaboration diagram for List:




### Public Methods

- DataValue head () throws Exceptional
- List tail ()
- List insert (DataValue dataValue)
- List append (DataValue dataValue)
- List concat (List list)

### 6.40.1 Member Function Documentation

#### 6.40.1.1 List List::append (DataValue *dataValue*)

Reimplemented in ListImpl.

Referenced by ListImpl::append().

#### 6.40.1.2 List List::concat (List *list*)

Reimplemented in ListImpl.

Referenced by ListImpl::concat().

#### 6.40.1.3 DataValue List::head ()

Reimplemented in ListImpl.

#### 6.40.1.4 List List::insert (DataValue *dataValue*)

Reimplemented in ListImpl.

Referenced by DataImpl::tupleToList().

#### 6.40.1.5 List List::tail ()

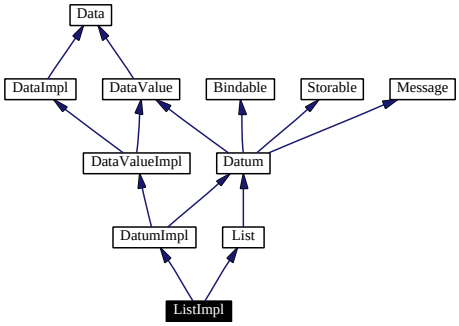
Reimplemented in ListImpl.

The documentation for this interface was generated from the following file:

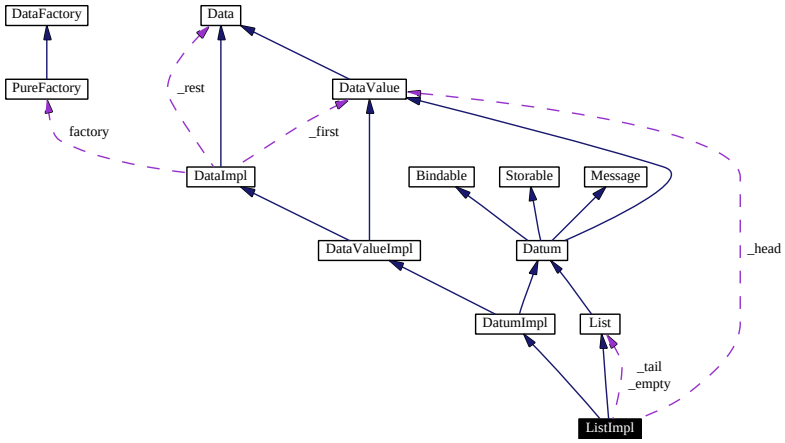
- List.java

## 6.41 ListImpl Class Reference

Inheritance diagram for ListImpl:




Collaboration diagram for ListImpl:




### Public Methods

- DataValue head () throws Exceptional
- List tail ()
- List insert (DataValue dataValue)
- List append (DataValue dataValue)
- List concat (List list)
- Empty equals (Data data) throws Exceptional
- String toString ()

### 6.41.1 Member Function Documentation

#### 6.41.1.1 List ListImpl::append (DataValue *dataValue*)

Reimplemented from List.

Definition at line 54 of file ListImpl.java.

#### 6.41.1.2 List ListImpl::concat (List *list*)

Reimplemented from List.

Definition at line 60 of file ListImpl.java.

#### 6.41.1.3 Empty ListImpl::equals (Data *data*)

Reimplemented from DataValueImpl.

Definition at line 68 of file ListImpl.java.

#### 6.41.1.4 DataValue ListImpl::head ()

Reimplemented from List.

Definition at line 40 of file ListImpl.java.

#### 6.41.1.5 List ListImpl::insert (DataValue *dataValue*)

Reimplemented from List.

Definition at line 50 of file ListImpl.java.

#### 6.41.1.6 List ListImpl::tail ()

Reimplemented from List.

Definition at line 46 of file ListImpl.java.

#### 6.41.1.7 String ListImpl::toString ()

Reimplemented from DataImpl.

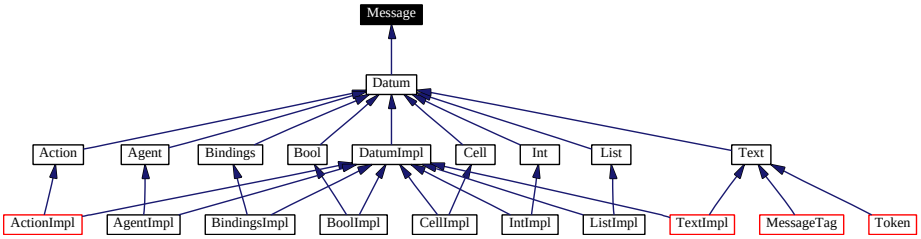
Definition at line 81 of file ListImpl.java.

The documentation for this class was generated from the following file:

- ListImpl.java

## 6.42 Message Interface Reference

Inheritance diagram for Message:

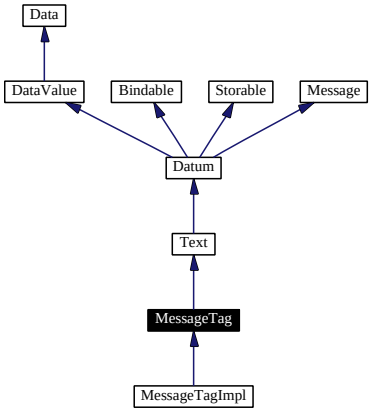



The documentation for this interface was generated from the following file:

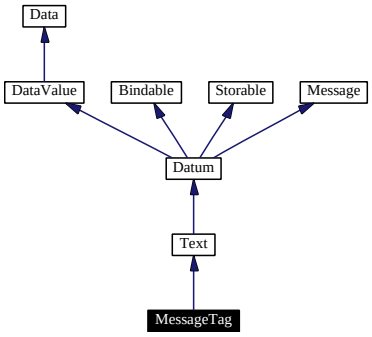
- Message.java

# 6.43 MessageTag Interface Reference

Inheritance diagram for MessageTag:




Collaboration diagram for MessageTag:

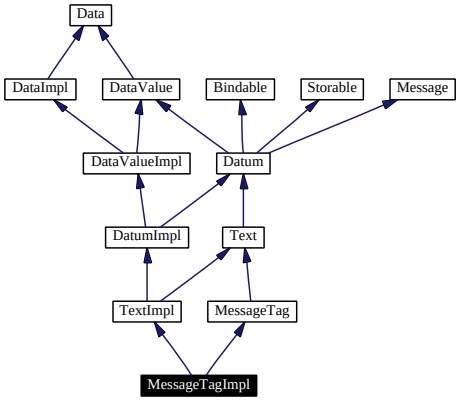



The documentation for this interface was generated from the following file:

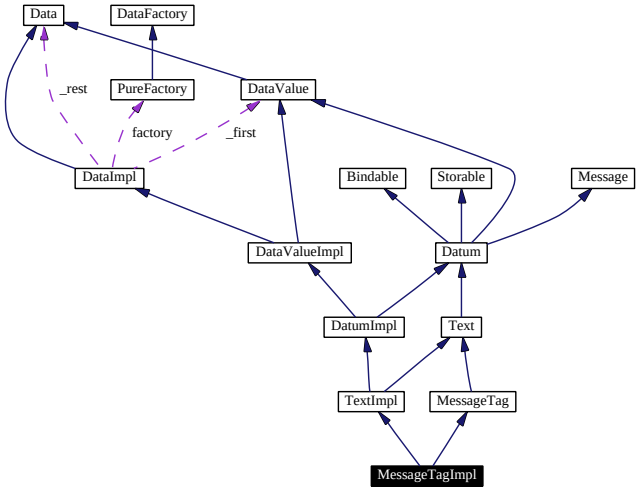
- MessageTag.java

## 6.44 MessageTagImpl Class Reference

Inheritance diagram for MessageTagImpl:




Collaboration diagram for MessageTagImpl:

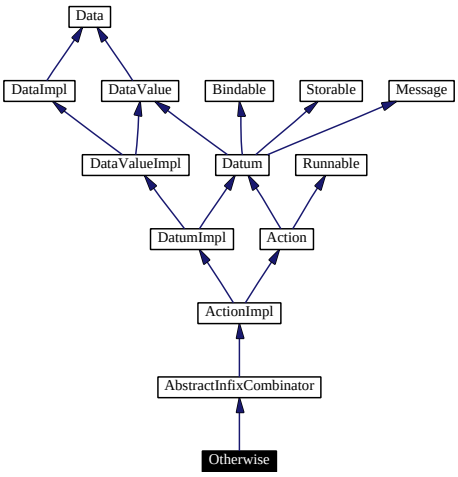



The documentation for this class was generated from the following file:

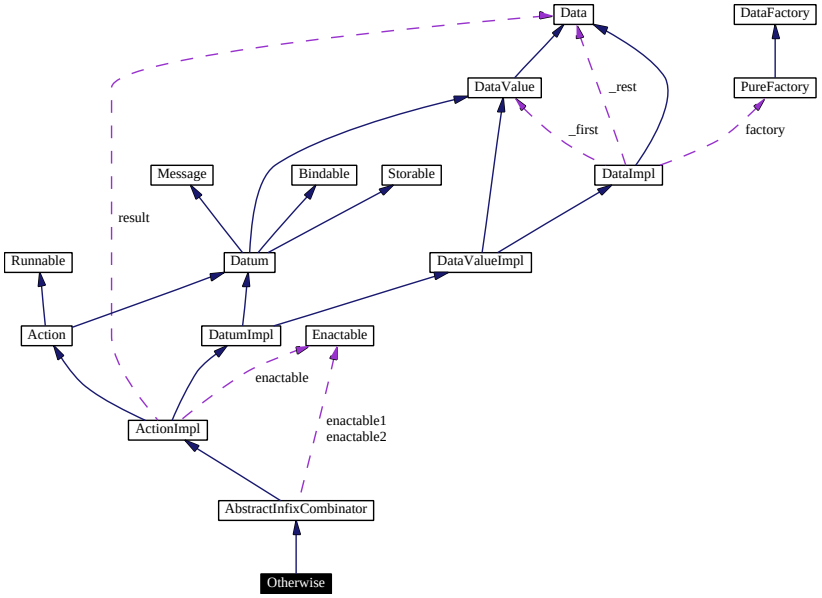
- MessageTagImpl.java

# 6.45 Otherwise Class Reference

Inheritance diagram for Otherwise:




Collaboration diagram for Otherwise:




## Public Methods

- Otherwise (PureFactory factory, Enactable enactable1, Enactable enactable2)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- Otherwise (PureFactory factory, Enactable enactable1, Enactable enactable2)

### 6.45.1 Constructor & Destructor Documentation

#### 6.45.1.1 Otherwise::Otherwise (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 131 of file AbstractAction.java.

#### 6.45.1.2 Otherwise::Otherwise (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 171 of file ActionImpl.java.

### 6.45.2 Member Function Documentation

#### 6.45.2.1 Data Otherwise::enact (Data *data*, Bindings *bindings*)

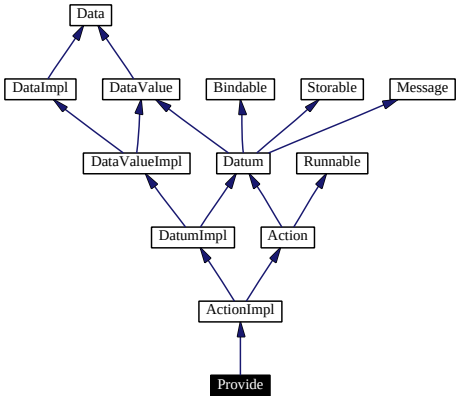
Definition at line 134 of file AbstractAction.java.

The documentation for this class was generated from the following files:

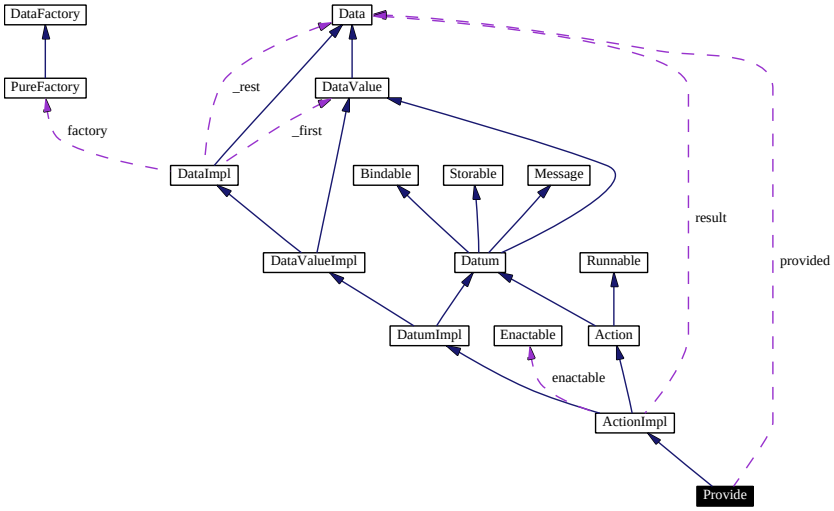
- AbstractAction.java
- ActionImpl.java

# 6.46 Provide Class Reference

Inheritance diagram for Provide:




Collaboration diagram for Provide:




## Public Methods

- Provide (PureFactory factory, Data data)

### 6.46.1 Constructor & Destructor Documentation

#### 6.46.1.1 Provide::Provide (PureFactory *factory*, Data *data*)

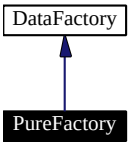
Definition at line 64 of file DataImpl.java.

The documentation for this class was generated from the following file:

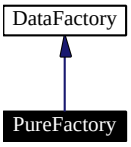
- DataImpl.java

## 6.47 PureFactory Class Reference

Inheritance diagram for PureFactory:




Collaboration diagram for PureFactory:




### Public Methods

- PureFactory ()
- Int makeInt (long n)
- Token makeToken (String s)
- Bool makeBool (boolean b)
- MessageTag makeMessageTag (String s)
- List makeList0 ()
- List makeList1 (DataValue dataValue)
- Bindings makeNoBindings ()
- Bindings makeBindings (Token token, Bindable bindable)
- Agent makeAgent ()
- Cell makeCell ()
- Empty makeEmpty ()
- Data makeTuple (DataValue dataValue, Data data)
- Action makeAction (Enactable enactable)

### 6.47.1 Constructor & Destructor Documentation

#### 6.47.1.1 PureFactory::PureFactory ()

Definition at line 9 of file PureFactory.java.

### 6.47.2 Member Function Documentation

#### 6.47.2.1 Action PureFactory::makeAction (Enactable *enactable*)

Reimplemented from DataFactory.

Definition at line 56 of file PureFactory.java.

#### 6.47.2.2 Agent PureFactory::makeAgent ()

Reimplemented from DataFactory.

Definition at line 42 of file PureFactory.java.

#### 6.47.2.3 Bindings PureFactory::makeBindings (Token *token*, Bindable *bindable*)

Reimplemented from DataFactory.

Definition at line 38 of file PureFactory.java.

#### 6.47.2.4 Bool PureFactory::makeBool (boolean *b*)

Reimplemented from DataFactory.

Definition at line 20 of file PureFactory.java.

Referenced by BoolImpl::not().

#### 6.47.2.5 Cell PureFactory::makeCell ()

Reimplemented from DataFactory.

Definition at line 45 of file PureFactory.java.

#### 6.47.2.6 Empty PureFactory::makeEmpty ()

Reimplemented from DataFactory.

Definition at line 48 of file PureFactory.java.

Referenced by EmptyImpl::at(), DataValueImpl::at(), BindingsImpl::bound(), ActionImpl::enact(), TextImpl::equals(), ListImpl::equals(), IntImpl::equals(), DataValueImpl::equals(), BoolImpl::equals(), ActionImpl::equals(), EmptyImpl::first(), IntImpl::greater(), IntImpl::greaterOrEq(), ListImpl::head(), IntImpl::less(), IntImpl::lessOrEq(), and DataValueImpl::rest().

#### 6.47.2.7 Int PureFactory::makeInt (long *n*)

Reimplemented from DataFactory.

Definition at line 14 of file PureFactory.java.

#### 6.47.2.8 List PureFactory::makeList0 ()

Reimplemented from DataFactory.

Definition at line 26 of file PureFactory.java.

Referenced by EmptyImpl::tupleToList().

#### 6.47.2.9 List PureFactory::makeList1 (DataValue *dataValue*)

Reimplemented from DataFactory.

Definition at line 30 of file PureFactory.java.

Referenced by DataValueImpl::tupleToList().

#### 6.47.2.10 MessageTag PureFactory::makeMessageTag (String *s*)

Reimplemented from DataFactory.

Definition at line 23 of file PureFactory.java.

#### 6.47.2.11 Bindings PureFactory::makeNoBindings ()

Reimplemented from DataFactory.

Definition at line 34 of file PureFactory.java.

Referenced by ActionImpl::enact().

#### 6.47.2.12 Token PureFactory::makeToken (String *s*)

Reimplemented from DataFactory.

Definition at line 17 of file PureFactory.java.

#### 6.47.2.13 Data PureFactory::makeTuple (DataValue *dataValue*, Data *data*)

Reimplemented from DataFactory.

Definition at line 52 of file PureFactory.java.

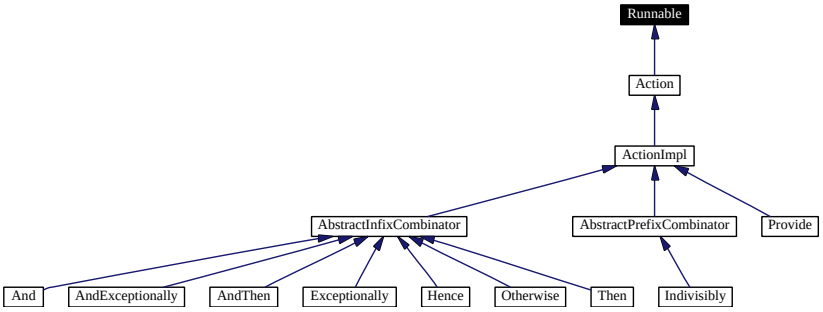
Referenced by DataValueImpl::concat().

The documentation for this class was generated from the following file:

- PureFactory.java

## 6.48 Runnable Class Reference

Inheritance diagram for Runnable:

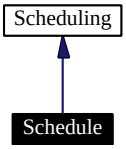



The documentation for this class was generated from the following file:

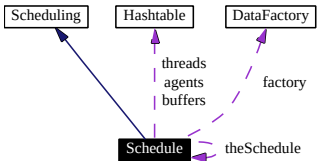
- Action.java

## 6.49 Schedule Class Reference

Inheritance diagram for Schedule:




Collaboration diagram for Schedule:




### Public Methods

- synchronized Empty send (Agent agent, Message message, MessageTag messageTag) throws Exceptional
- synchronized Message receive (MessageTag messageTag) throws Exceptional
- synchronized Agent activate (Action action)
- synchronized Empty deactivate (Agent agent) throws Exceptional
- synchronized Agent currentAgent ()
- Int currentTime ()
- Int chooseNatural ()

### Static Public Methods

- void initSchedule (DataFactory factory)
- Schedule getSchedule ()

### 6.49.1 Member Function Documentation

#### 6.49.1.1 synchronized Agent Schedule::activate (Action *action*)

Reimplemented from Scheduling.

Definition at line 60 of file Schedule.java.

#### 6.49.1.2 Int Schedule::chooseNatural ()

Reimplemented from Scheduling.

Definition at line 90 of file Schedule.java.

#### 6.49.1.3 synchronized Agent Schedule::currentAgent ()

Reimplemented from Scheduling.

Definition at line 81 of file Schedule.java.

#### 6.49.1.4 Int Schedule::currentTime ()

Reimplemented from Scheduling.

Definition at line 85 of file Schedule.java.

#### 6.49.1.5 synchronized Empty Schedule::deactivate (Agent *agent*)

Reimplemented from Scheduling.

Definition at line 70 of file Schedule.java.

#### 6.49.1.6 Schedule Schedule::getSchedule () [static]

Definition at line 32 of file Schedule.java.

#### 6.49.1.7 void Schedule::initSchedule (DataFactory *factory*) [static]

Definition at line 28 of file Schedule.java.

#### 6.49.1.8 synchronized Message Schedule::receive (MessageTag *messageTag*)

Reimplemented from Scheduling.

Definition at line 48 of file Schedule.java.

#### 6.49.1.9 synchronized Empty Schedule::send (Agent *agent*, Message *message*, MessageTag *messageTag*)

Reimplemented from Scheduling.

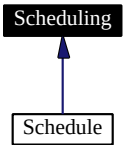
Definition at line 40 of file Schedule.java.

The documentation for this class was generated from the following file:

- Schedule.java

## 6.50 Scheduling Interface Reference

Inheritance diagram for Scheduling:




### Public Methods

- Empty send (Agent agent, Message message, MessageTag messageTag) throws Exceptional
- Message receive (MessageTag messageTag) throws Exceptional
- Agent activate (Action action)
- Empty deactivate (Agent agent) throws Exceptional
- Agent currentAgent ()
- Int currentTime ()
- Int chooseNatural ()

### 6.50.1 Member Function Documentation

#### 6.50.1.1 Agent Scheduling::activate (Action *action*)

Reimplemented in Schedule.

#### 6.50.1.2 Int Scheduling::chooseNatural ()

Reimplemented in Schedule.

#### 6.50.1.3 Agent Scheduling::currentAgent ()

Reimplemented in Schedule.

#### 6.50.1.4 Int Scheduling::currentTime ()

Reimplemented in Schedule.

#### 6.50.1.5 Empty Scheduling::deactivate (Agent *agent*)

Reimplemented in Schedule.

#### 6.50.1.6 Message Scheduling::receive (MessageTag *messageTag*)

Reimplemented in Schedule.

#### 6.50.1.7 Empty Scheduling::send (Agent *agent*, Message *message*, MessageTag *messageTag*)

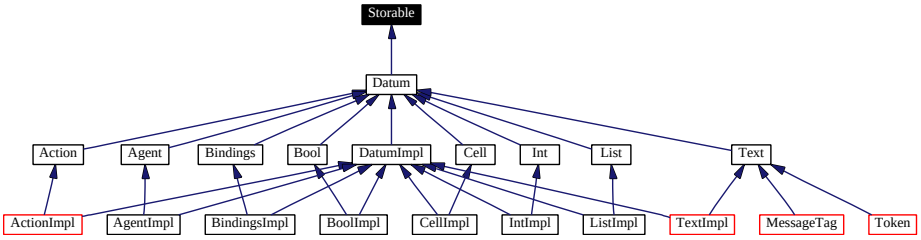
Reimplemented in Schedule.

The documentation for this interface was generated from the following file:

- Scheduling.java

## 6.51 Storable Interface Reference

Inheritance diagram for Storable:

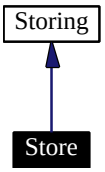



The documentation for this interface was generated from the following file:

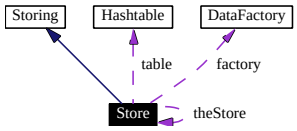
- Storable.java

# 6.52 Store Class Reference

Inheritance diagram for Store:




Collaboration diagram for Store:




## Public Methods

- synchronized Cell create (Storable o)
- synchronized Empty destroy (Cell cell) throws Exceptional
- synchronized Empty update (Cell cell, Storable o) throws Exceptional
- synchronized Storable inspect (Cell cell) throws Exceptional
- String toString ()

## Static Public Methods

- void initStore (DataFactory factory)
- void initStore (DataFactory factory, int cap)
- void initStore (DataFactory factory, int cap, int loadf)
- Store getStore ()

## 6.52.1 Member Function Documentation

### 6.52.1.1 synchronized Cell Store::create (Storable *o*)

Reimplemented from Storing.

Definition at line 37 of file Store.java.

### 6.52.1.2 synchronized Empty Store::destroy (Cell *cell*)

Reimplemented from Storing.

Definition at line 43 of file Store.java.

#### 6.52.1.3 Store Store::getStore () [static]

Definition at line 30 of file Store.java.

#### 6.52.1.4 void Store::initStore (DataFactory *factory*, int *cap*, int *loadf*) [static]

Definition at line 26 of file Store.java.

#### 6.52.1.5 void Store::initStore (DataFactory *factory*, int *cap*) [static]

Definition at line 22 of file Store.java.

#### 6.52.1.6 void Store::initStore (DataFactory *factory*) [static]

Definition at line 18 of file Store.java.

#### 6.52.1.7 synchronized Storable Store::inspect (Cell *cell*)

Reimplemented from Storing.

Definition at line 56 of file Store.java.

#### 6.52.1.8 String Store::toString ()

Definition at line 62 of file Store.java.

#### 6.52.1.9 synchronized Empty Store::update (Cell *cell*, Storable *o*)

Reimplemented from Storing.

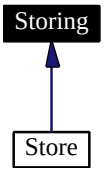
Definition at line 49 of file Store.java.

The documentation for this class was generated from the following file:

- Store.java

# 6.53 Storing Interface Reference

Inheritance diagram for Storing:




## Public Methods

- Cell create (Storable storable)
- Empty destroy (Cell cell) throws Exceptional
- Empty update (Cell cell, Storable o) throws Exceptional
- Storable inspect (Cell cell) throws Exceptional

## 6.53.1 Member Function Documentation

### 6.53.1.1 Cell Storing::create (Storable *o*)

Reimplemented in Store.

### 6.53.1.2 Empty Storing::destroy (Cell *cell*)

Reimplemented in Store.

### 6.53.1.3 Storable Storing::inspect (Cell *cell*)

Reimplemented in Store.

### 6.53.1.4 Empty Storing::update (Cell *cell*, Storable *o*)

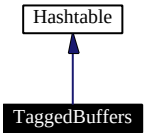
Reimplemented in Store.

The documentation for this interface was generated from the following file:

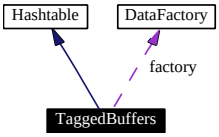
- Storing.java

## 6.54 TaggedBuffers Class Reference

Inheritance diagram for TaggedBuffers:




Collaboration diagram for TaggedBuffers:




### Public Methods

- TaggedBuffers (DataFactory factory)
- void queue (MessageTag messageTag, Message message)
- Message dequeue (MessageTag messageTag) throws Exceptional

### 6.54.1 Constructor & Destructor Documentation

#### 6.54.1.1 TaggedBuffers::TaggedBuffers (DataFactory *factory*)

Definition at line 11 of file TaggedBuffers.java.

### 6.54.2 Member Function Documentation

#### 6.54.2.1 Message TaggedBuffers::dequeue (MessageTag *messageTag*)

Definition at line 26 of file TaggedBuffers.java.

Referenced by Schedule::receive().

#### 6.54.2.2 void TaggedBuffers::queue (MessageTag *messageTag*, Message *message*)

Definition at line 16 of file TaggedBuffers.java.

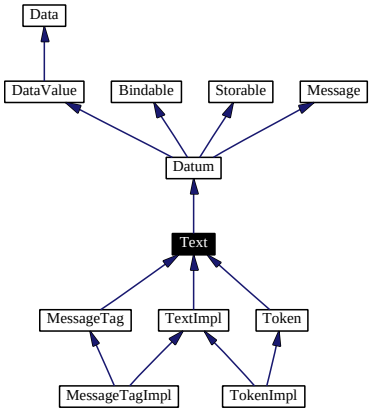
Referenced by Schedule::send().

The documentation for this class was generated from the following file:

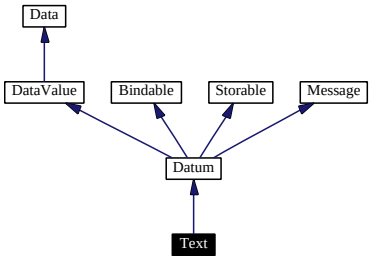
- TaggedBuffers.java

# 6.55 Text Interface Reference

Inheritance diagram for Text:




Collaboration diagram for Text:




## Public Methods

- String stringValue ()

## 6.55.1 Member Function Documentation

### 6.55.1.1 String Text::stringValue ()

Reimplemented in TextImpl.

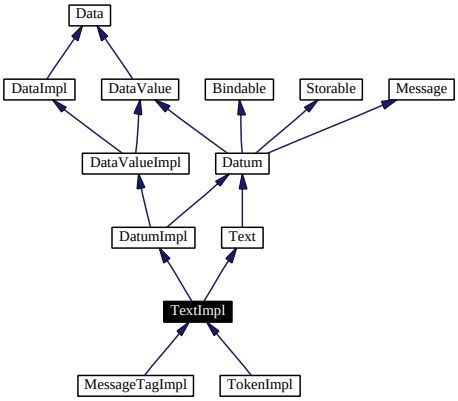
Referenced by TokenImpl::compareTo(), and TokenImpl::toString().

The documentation for this interface was generated from the following file:

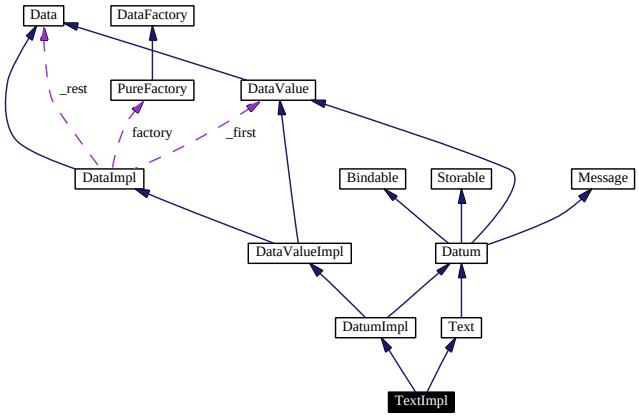
- Text.java

## 6.56 TextImpl Class Reference

Inheritance diagram for TextImpl:




Collaboration diagram for TextImpl:




### Public Methods

- String stringValue ()
- Empty equals (Data data) throws Exceptional

### 6.56.1 Member Function Documentation

#### 6.56.1.1 Empty TextImpl::equals (Data *data*)

Reimplemented from DataValueImpl.

Definition at line 19 of file TextImpl.java.

#### 6.56.1.2 String TextImpl::stringValue ()

Reimplemented from Text.

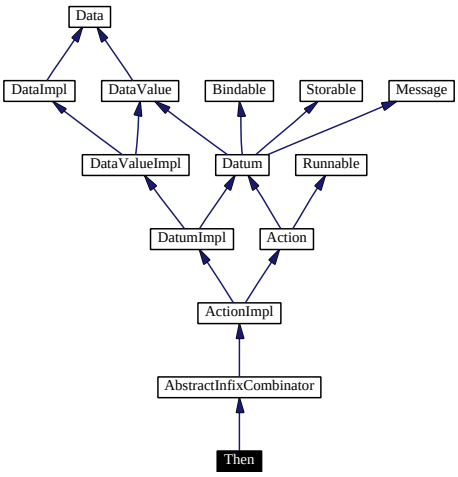
Definition at line 14 of file TextImpl.java.

The documentation for this class was generated from the following file:

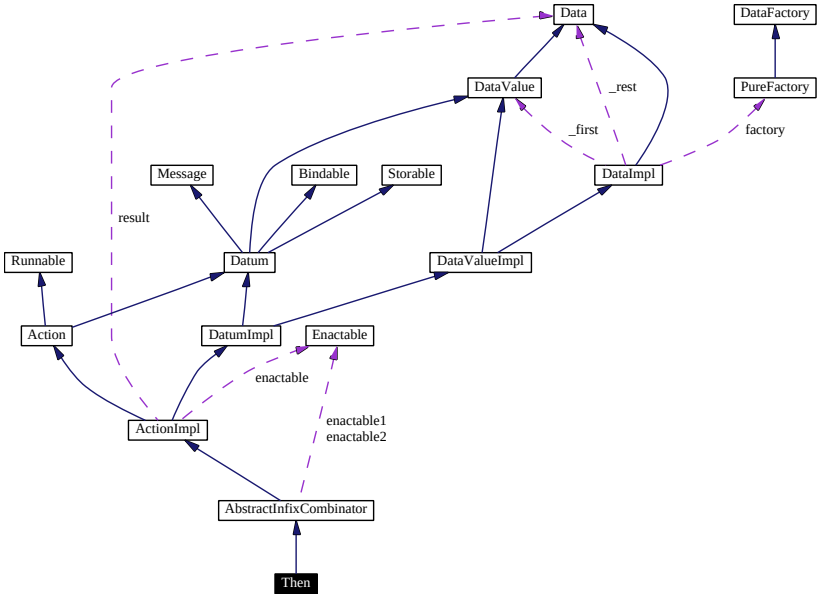
- TextImpl.java

## 6.57 Then Class Reference

Inheritance diagram for Then:




Collaboration diagram for Then:




### Public Methods

- Then (PureFactory factory, Enactable enactable1, Enactable enactable2)
- Data enact (Data data, Bindings bindings) throws Exceptional, Failed
- Then (PureFactory factory, Enactable enactable1, Enactable enactable2)

## 6.57.1 Constructor & Destructor Documentation

### 6.57.1.1 Then::Then (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 64 of file AbstractAction.java.

### 6.57.1.2 Then::Then (PureFactory *factory*, Enactable *enactable1*, Enactable *enactable2*)

Definition at line 84 of file ActionImpl.java.

## 6.57.2 Member Function Documentation

### 6.57.2.1 Data Then::enact (Data *data*, Bindings *bindings*)

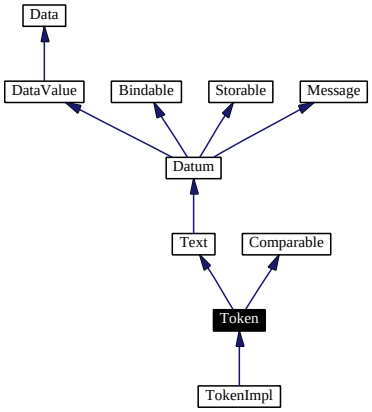
Definition at line 67 of file AbstractAction.java.

The documentation for this class was generated from the following files:

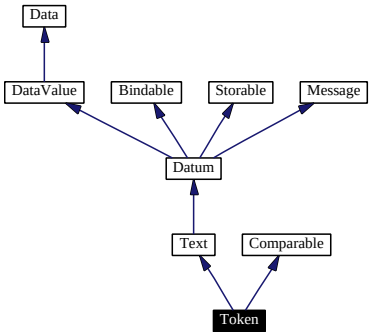
- AbstractAction.java
- ActionImpl.java

## 6.58 Token Interface Reference

Inheritance diagram for Token:




Collaboration diagram for Token:

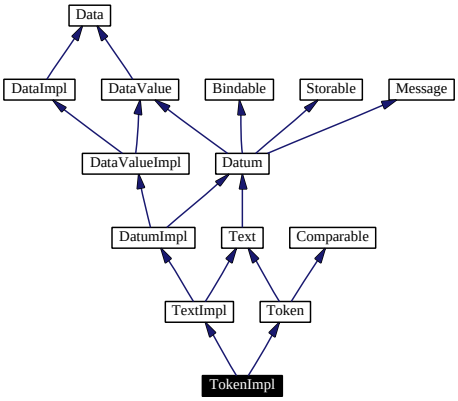



The documentation for this interface was generated from the following file:

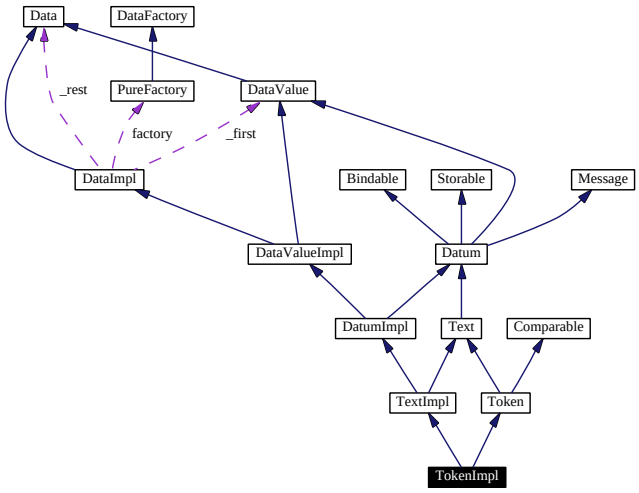
- Token.java

# 6.59 TokenImpl Class Reference

Inheritance diagram for TokenImpl:




Collaboration diagram for TokenImpl:




## Public Methods

- int compareTo (Object o)
- String toString ()

## 6.59.1 Member Function Documentation

### 6.59.1.1 int TokenImpl::compareTo (Object *o*)

Definition at line 12 of file TokenImpl.java.

#### 6.59.1.2 String TokenImpl::toString ()

Reimplemented from DataImpl.

Definition at line 16 of file TokenImpl.java.

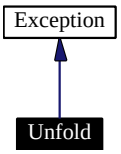
The documentation for this class was generated from the following file:

- TokenImpl.java

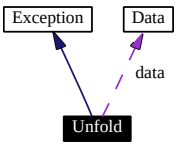
# 6.60 Unfold Class Reference

Describe class `Unfold` here.

Inheritance diagram for Unfold:




Collaboration diagram for Unfold:




## Public Methods

- Unfold (Data data)

  *Creates a new* `Unfold` *instance.*

- Data getData ()

  *Describe* `getData` *method here.*

## Static Public Methods

- Data unfold (Data data) throws Unfold

  *Describe* `unfold` *method here.*

### 6.60.1 Detailed Description

Describe class `Unfold` here.

**Author:**
`Tijs van der Storm`

**Version:**
1.0

Definition at line 12 of file Unfold.java.

## 6.60.2 Constructor & Destructor Documentation

### 6.60.2.1 Unfold::Unfold (Data *data*)

Creates a new `Unfold` instance.

**Parameters:**
***data*** a `Data` value

Definition at line 32 of file Unfold.java.

## 6.60.3 Member Function Documentation

### 6.60.3.1 Data Unfold::getData ()

Describe `getData` method here.

**Returns:**
a `Data` value

Definition at line 41 of file Unfold.java.

### 6.60.3.2 Data Unfold::unfold (Data *data*) `[static]`

Describe `unfold` method here.

**Parameters:**
***data*** given `Data`

**Returns:**
nothing

**Exceptions:**
***Unfold*** is always thrown with the given data

Definition at line 22 of file Unfold.java.

The documentation for this class was generated from the following file:

- Unfold.java

# Chapter 7

# Action Java Compiler Runtime File Documentation

## 7.1 AbstractAction.java File Reference

### Compounds

- class ActionImpl
- class And
- class AndExceptionally
- class AndThen
- class Exceptionally
- class Hence
- class Indivisibly
- class Otherwise
- class Then

## 7.2 AbstractEnactable.java File Reference

### Compounds

- class AbstractEnactable

# 7.3 AbstractInfixCombinator.java File Reference

## Compounds

- class AbstractInfixCombinator

## 7.4 AbstractPrefixCombinator.java File Reference

### Compounds

- class AbstractPrefixCombinator

# 7.5 Action.java File Reference

## Compounds

- interface Action

## 7.6 ActionImpl.java File Reference

### Compounds

- class **And::_And**
- class **AndExceptionally::_AndExceptionally**
- class **AndThen::_AndThen**
- class **Exceptionally::_Exceptionally**
- class **Hence::_Hence**
- class **Indivisibly::_Indivisibly**
- class **Otherwise::_Otherwise**
- class **Then::_Then**
- class ActionImpl
- class And
- class AndExceptionally
- class AndThen
- class Exceptionally
- class Hence
- class Indivisibly
- class Otherwise
- class Then

# 7.7 Agent.java File Reference

## Compounds

- interface Agent

## 7.8 AgentImpl.java File Reference

### Compounds

- class AgentImpl

# 7.9 Bindable.java File Reference

## Compounds

- interface Bindable

## 7.10 Bindings.java File Reference

### Compounds

- interface Bindings

# 7.11 BindingsImpl.java File Reference

## Compounds

- class BindingsImpl

## 7.12 Bool.java File Reference

### Compounds

- interface Bool

# 7.13 BoolImpl.java File Reference

## Compounds

- class BoolImpl

## 7.14 Cell.java File Reference

### Compounds

- interface Cell

# 7.15 CellImpl.java File Reference

## Compounds

- class CellImpl

## 7.16 Data.java File Reference

### Compounds

- interface Data

# 7.17 DataConst.java File Reference

## Compounds

- interface DataConst

## 7.18 DataConstImpl.java File Reference

### Compounds

- class DataConstImpl

# 7.19 DataFactory.java File Reference

## Compounds

- interface DataFactory

## 7.20 DataImpl.java File Reference

### Compounds

- class **Provide::_Provide**
- class DataImpl
- class Provide

# 7.21 DataValue.java File Reference

## Compounds

- interface DataValue

## 7.22 DataValueImpl.java File Reference

### Compounds

- class DataValueImpl

# 7.23 Datum.java File Reference

## Compounds

- interface Datum

## 7.24 DatumImpl.java File Reference

### Compounds

- class DatumImpl

# 7.25 Empty.java File Reference

## Compounds

- interface Empty

## 7.26 EmptyImpl.java File Reference

### Compounds

- class EmptyImpl

# 7.27 Enactable.java File Reference

## Compounds

- interface Enactable

## 7.28 Exceptional.java File Reference

### Compounds

- class Exceptional

# 7.29 Failed.java File Reference

## Compounds

- class Failed

## 7.30 Int.java File Reference

### Compounds

- interface Int

# 7.31 IntImpl.java File Reference

## Compounds

- class IntImpl

## 7.32 List.java File Reference

### Compounds

- interface List

# 7.33 ListImpl.java File Reference

## Compounds

- class ListImpl

## 7.34 Message.java File Reference

### Compounds

- interface Message

# 7.35 MessageTag.java File Reference

## Compounds

- interface MessageTag

## 7.36 MessageTagImpl.java File Reference

### Compounds

- class MessageTagImpl

# 7.37 PureFactory.java File Reference

## Compounds

- class PureFactory

## 7.38 Schedule.java File Reference

### Compounds

- class Schedule

# 7.39 Scheduling.java File Reference

## Compounds

- interface Scheduling

## 7.40 Storable.java File Reference

### Compounds

- interface Storable

# 7.41 Store.java File Reference

## Compounds

- class Store

## 7.42 Storing.java File Reference

### Compounds

- interface Storing

# 7.43 TaggedBuffers.java File Reference

## Compounds

- class TaggedBuffers

## 7.44 Text.java File Reference

### Compounds

- interface Text

# 7.45 TextImpl.java File Reference

## Compounds

- class TextImpl

## 7.46 Token.java File Reference

### Compounds

- interface Token

# 7.47 TokenImpl.java File Reference

## Compounds

- class TokenImpl

## 7.48 Tuple.java File Reference

# 7.49 Unfold.java File Reference

## Compounds

- class Unfold

  *Describe class* `Unfold` *here.*