XML Metadata Interchange (XMI) Appendices

Proposal to the OMG OA&DTF RFP 3:
Stream-based Model Interchange Format (SMIF)

Joint Submission

Unisys Corporation	***Fujitsu***
International Business Machines Corporation	***Softeam***
Cooperative Research Centre for Distributed Systems Technology (DSTC)	***Recerca Informatica***
Oracle Corporation	***Daimler-Benz***
Platinum Technology, Inc.	

Supported by:

Cayenne Software	*Ardent*
Genesis Development	*Aviatis*
Inline Software	*ICONIX*
Rational Software Corporation	*Integrated Systems*
Select Software	*Verilog*
Sprint Communications Company	*Telefonica I+D*
Sybase, Inc.	*Universitat Politecnica de Catalunya*
Xerox	*NCR*
MCI Systemhouse	*Nihon Unisys*
Boeing	*NTT*

OMG Document ad/98-10-06
October 20, 1998

Copyright 1998, Unisys Corporation
Copyright 1998, IBM Corporation
Copyright 1998, Cooperative Research Centre for Distributed Systems Technology (DSTC)
Copyright 1998, Oracle Corporation
Copyright 1998, Platinum Technology, Inc.
Copyright 1998, Fujitsu
Copyright 1998, Softeam
Copyright 1998, Recerca Informatica
Copyright 1998, Daimler-Benz

The companies listed above hereby grant a royalty-free license to the Object Management Group, Inc. (OMG) for worldwide distribution of this document or any derivative works thereof, so long as the OMG reproduces the copyright notices and the below paragraphs on all distributed copies.

The material in this document is submitted to the OMG for evaluation. Submission of this document does not represent a commitment to implement any portion of this specification in the products of the submitters.

WHILE THE INFORMATION IN THIS PUBLICATION IS BELIEVED TO BE ACCURATE,THE COMPANIES LISTED ABOVE MAKE NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. The companies listed above shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material. The information contained in this document is subject to change without notice.

This document contains information which is protected by copyright. All Rights Reserved. Except as otherwise provided herein, no part of this work may be reproduced or used in any form or by any meansógraphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systemsó without the permission of one of the copyright owners. All copies of this document must include the copyright and other information contained on this page.

The copyright owners grant member companies of the OMG permission to make a limited number of copies of this document (up to fifty copies) for their internal use as part of the OMG evaluation process.

RESTRICTED RIGHTS LEGEND. Use, duplication, or disclosure by government is subject to restrictions as set forth in subdivision (c) (1) (ii) of the Right in Technical Data and Computer Software Clause at DFARS 252.227.7013.

CORBA, OMG, and Object Request Broker are trademarks of Object Management Group.

Table of Contents

Appendix A UML 1.1 DTD A-5

A.1 Introduction A-5
A.2 UML DTD A-5

Appendix B MOF 1.1 DTD B-193

B.1 Introduction B-193
B.2 MOF DTD B-193
B.3 Example B-214

Appendix C Example Model Encodings C-217

C.1 Introduction C-217
C.2 Model1 Model C-217
C.3 Business Model C-219

UML 1.1 DTD A

A.1 Introduction

This appendix contains a normative DTD that represents the UML 1.1 metamodel. It was generated by a program that closely follows rule set 1 in Chapter 7, "XML DTD Production". The program created the DTD from a Rational Rose model representing the UML 1.1 metamodel as defined in the document "UML Semantics version 1.1, 1 September 1997". The metamodel in that document was changed by adding role names to associations that did not have them.

As noted in Section 4.4.12 of the XMI submission, the XMI submitters are not attempting to create official metamodels for UML (or for the MOF). We propose that the groups that propose and maintain the standards for metamodels create official versions of the metamodels and DTDs corresponding to them.

In the meantime, there is a need to exchange MOF and UML metadata now, so this appendix presents a normative DTD for UML 1.1; it can be used to develop software.

On minor change from rule set 1 is that all composed roles are included at the beginning of the UML declarations in the DTD, whereas rule set 1 would place them in the packages that they belong to. That difference does not change the semantics of the DTD.

The source of the UML 1.1 metamodel may have outstanding issues that might affect the DTD included in this appendix.

A.2 UML DTD

```
<?xml version="1.0" encoding="UTF-8" ?>

<!-- XMI Automatic DTD Generation        -->
<!-- Date: Fri Oct 16 14:42:45 PDT 1998 -->
<!-- Metamodel: uml11                   -->
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI is the top-level XML element for XMI transfer text               -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI  (XMI.header, XMI.content?, XMI.difference*,
                   XMI.extensions*) >
<!ATTLIST XMI
              xmi.version CDATA #FIXED "1.0"
              timestamp CDATA #IMPLIED
              verified (true | false) #IMPLIED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.header contains documentation and identifies the model,          -->
<!-- metamodel, and metametamodel                                         -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.header (XMI.documentation?, XMI.model*, XMI.metamodel*,
                          XMI.metametamodel*) >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- documentation for transfer data                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.documentation (#PCDATA | XMI.owner | XMI.contact |
                                  XMI.longDescription | XMI.shortDescription |
                                  XMI.exporter | XMI.exporterVersion |
                                  XMI.notice)* >

<!ELEMENT XMI.owner ANY >

<!ELEMENT XMI.contact ANY >

<!ELEMENT XMI.longDescription ANY >

<!ELEMENT XMI.shortDescription ANY >

<!ELEMENT XMI.exporter ANY >

<!ELEMENT XMI.exporterVersion ANY >

<!ELEMENT XMI.exporterID ANY >

<!ELEMENT XMI.notice ANY >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.element.att defines the attributes that each XML element         -->
<!-- that corresponds to a metamodel class must have to conform to        -->
<!-- the XMI specification.                                               -->
<!-- ____________________________________________________________________ -->
```

```
<!ENTITY % XMI.element.att
                  'xmi.id ID #IMPLIED xmi.label CDATA #IMPLIED xmi.uuid
                   CDATA #IMPLIED ' >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.link.att defines the attributes that each XML element that  -->
<!-- corresponds to a metamodel class must have to enable it to       -->
<!-- function as a simple XLink as well as refer to model              -->
<!-- constructs within the same XMI file.                                -->
<!-- ____________________________________________________________________ -->

<!ENTITY % XMI.link.att
                  'xml:link CDATA #IMPLIED inline (true | false) #IMPLIED
                   actuate (show | user) #IMPLIED href CDATA #IMPLIED role
                   CDATA #IMPLIED title CDATA #IMPLIED show (embed | replace
                   | new) #IMPLIED behavior CDATA #IMPLIED xmi.idref IDREF
                   #IMPLIED xmi.uuidref CDATA #IMPLIED' >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.model identifies the model(s) being transferred               -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.model ANY >
<!ATTLIST XMI.model
              %XMI.link.att;
              xmi.name      CDATA #REQUIRED
              xmi.version   CDATA #IMPLIED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.metamodel identifies the metamodel(s) for the transferred   -->
<!-- data                                                                 -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.metamodel ANY >
<!ATTLIST XMI.metamodel
              %XMI.link.att;
              xmi.name      CDATA #REQUIRED
              xmi.version   CDATA #IMPLIED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.metametamodel identifies the metametamodel(s) for the        -->
<!-- transferred data                                                     -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.metametamodel ANY >
<!ATTLIST XMI.metametamodel
              %XMI.link.att;
              xmi.name      CDATA #REQUIRED
              xmi.version   CDATA #IMPLIED
```

```
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.content is the actual data being transferred                     -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.content ANY >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.extensions contains data to transfer that does not conform  -->
<!-- to the metamodel(s) in the header                                    -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.extensions ANY >
<!ATTLIST XMI.extensions
              xmi.extender CDATA #REQUIRED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- extension contains information related to a specific model       -->
<!-- construct that is not defined in the metamodel(s) in the header -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.extension ANY >
<!ATTLIST XMI.extension
              %XMI.element.att;
              %XMI.link.att;
              xmi.extender    CDATA #REQUIRED
              xmi.extenderID CDATA #REQUIRED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.difference holds XML elements representing differences to a -->
<!-- base model                                                           -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.difference (XMI.difference | XMI.delete | XMI.add |
                          XMI.replace)* >
<!ATTLIST XMI.difference
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.delete represents a deletion from a base model                   -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.delete EMPTY >
<!ATTLIST XMI.delete
              %XMI.element.att;
```

```
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.add represents an addition to a base model                       -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.add ANY >
<!ATTLIST XMI.add
              %XMI.element.att;
              %XMI.link.att;
              xmi.position CDATA "-1"
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.replace represents the replacement of a model construct          -->
<!-- with another model construct in a base model                         -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.replace ANY >
<!ATTLIST XMI.replace
              %XMI.element.att;
              %XMI.link.att;
              xmi.position CDATA "-1"
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.reference may be used to refer to data types not defined in      -->
<!-- the metamodel                                                        -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.reference ANY >
<!ATTLIST XMI.reference
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- This section contains the declaration of XML elements                -->
<!-- representing data types                                              -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.TypeDefinitions ANY >

<!ELEMENT XMI.field ANY >

<!ELEMENT XMI.seqItem ANY >

<!ELEMENT XMI.octetStream (#PCDATA) >

<!ELEMENT XMI.unionDiscrim ANY >
```

```
<!ELEMENT XMI.enum EMPTY >
<!ATTLIST XMI.enum
              xmi.value CDATA #REQUIRED
>

<!ELEMENT XMI.any ANY >
<!ATTLIST XMI.any
              %XMI.link.att;
              xmi.type CDATA #IMPLIED
              xmi.name CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTypeCode (XMI.CorbaTcAlias | XMI.CorbaTcStruct |
                                  XMI.CorbaTcSequence | XMI.CorbaTcArray |
                                  XMI.CorbaTcEnum | XMI.CorbaTcUnion |
                                  XMI.CorbaTcExcept | XMI.CorbaTcString |
                                  XMI.CorbaTcWstring | XMI.CorbaTcShort |
                                  XMI.CorbaTcLong | XMI.CorbaTcUshort |
                                  XMI.CorbaTcUlong | XMI.CorbaTcFloat |
                                  XMI.CorbaTcDouble | XMI.CorbaTcBoolean |
                                  XMI.CorbaTcChar | XMI.CorbaTcWchar |
                                  XMI.CorbaTcOctet | XMI.CorbaTcAny |
                                  XMI.CorbaTcTypeCode | XMI.CorbaTcPrincipal |
                                  XMI.CorbaTcNull | XMI.CorbaTcVoid |
                                  XMI.CorbaTcLongLong |
                                  XMI.CorbaTcLongDouble) >
<!ATTLIST XMI.CorbaTypeCode
              %XMI.element.att;
>

<!ELEMENT XMI.CorbaTcAlias (XMI.CorbaTypeCode) >
<!ATTLIST XMI.CorbaTcAlias
              xmi.tcName CDATA #REQUIRED
              xmi.tcId   CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTcStruct (XMI.CorbaTcField)* >
<!ATTLIST XMI.CorbaTcStruct
              xmi.tcName CDATA #REQUIRED
              xmi.tcId   CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTcField (XMI.CorbaTypeCode) >
<!ATTLIST XMI.CorbaTcField
              xmi.tcName CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcSequence (XMI.CorbaTypeCode |
                                     XMI.CorbaRecursiveType) >
<!ATTLIST XMI.CorbaTcSequence
              xmi.tcLength CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaRecursiveType EMPTY >
<!ATTLIST XMI.CorbaRecursiveType
```

```
                xmi.offset  CDATA  #REQUIRED
>

<!ELEMENT XMI.CorbaTcArray (XMI.CorbaTypeCode) >
<!ATTLIST XMI.CorbaTcArray
                xmi.tcLength  CDATA  #REQUIRED
>

<!ELEMENT XMI.CorbaTcObjRef EMPTY >
<!ATTLIST XMI.CorbaTcObjRef
                xmi.tcName  CDATA  #REQUIRED
                xmi.tcId    CDATA  #IMPLIED
>

<!ELEMENT XMI.CorbaTcEnum (XMI.CorbaTcEnumLabel) >
<!ATTLIST XMI.CorbaTcEnum
                xmi.tcName  CDATA  #REQUIRED
                xmi.tcId    CDATA  #IMPLIED
>

<!ELEMENT XMI.CorbaTcEnumLabel EMPTY >
<!ATTLIST XMI.CorbaTcEnumLabel
                xmi.tcName  CDATA  #REQUIRED
>

<!ELEMENT XMI.CorbaTcUnionMbr (XMI.CorbaTypeCode, XMI.any) >
<!ATTLIST XMI.CorbaTcUnionMbr
                xmi.tcName  CDATA  #REQUIRED
>

<!ELEMENT XMI.CorbaTcUnion (XMI.CorbaTypeCode, XMI.CorbaTcUnionMbr*) >
<!ATTLIST XMI.CorbaTcUnion
                xmi.tcName  CDATA  #REQUIRED
                xmi.tcId    CDATA  #IMPLIED
>

<!ELEMENT XMI.CorbaTcExcept (XMI.CorbaTcField)* >
<!ATTLIST XMI.CorbaTcExcept
                xmi.tcName  CDATA  #REQUIRED
                xmi.tcId    CDATA  #IMPLIED
>

<!ELEMENT XMI.CorbaTcString EMPTY >
<!ATTLIST XMI.CorbaTcString
                xmi.tcLength  CDATA  #REQUIRED
>

<!ELEMENT XMI.CorbaTcWstring EMPTY >
<!ATTLIST XMI.CorbaTcWstring
                xmi.tcLength  CDATA  #REQUIRED
>

<!ELEMENT XMI.CorbaTcFixed EMPTY >
<!ATTLIST XMI.CorbaTcFixed
                xmi.tcDigits  CDATA  #REQUIRED
```

```
              xmi.tcScale   CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcShort EMPTY >

<!ELEMENT XMI.CorbaTcLong EMPTY >

<!ELEMENT XMI.CorbaTcUshort EMPTY >

<!ELEMENT XMI.CorbaTcUlong EMPTY >

<!ELEMENT XMI.CorbaTcFloat EMPTY >

<!ELEMENT XMI.CorbaTcDouble EMPTY >

<!ELEMENT XMI.CorbaTcBoolean EMPTY >

<!ELEMENT XMI.CorbaTcChar EMPTY >

<!ELEMENT XMI.CorbaTcWchar EMPTY >

<!ELEMENT XMI.CorbaTcOctet EMPTY >

<!ELEMENT XMI.CorbaTcAny EMPTY >

<!ELEMENT XMI.CorbaTcTypeCode EMPTY >

<!ELEMENT XMI.CorbaTcPrincipal EMPTY >

<!ELEMENT XMI.CorbaTcNull EMPTY >

<!ELEMENT XMI.CorbaTcVoid EMPTY >

<!ELEMENT XMI.CorbaTcLongLong EMPTY >

<!ELEMENT XMI.CorbaTcLongDouble EMPTY >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL:  uml11                                                    -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Namespace.ownedElement (Foundation.Core.ModelElement |
                                                  Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |
                                                  Foundation.Core.Classifier
|
                                                  Foundation.Core.Interface |
                                                      Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                   Foundation.Core.DataType |
```

```
Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
                                                    Model_Management.Subsystem
|

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |
                                                  Foundation.Core.Association
|

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |
                                                   Model_Management.Package |
                                                     Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |
                                                     Foundation.Core.Feature |

Foundation.Core.StructuralFeature |
                                                    Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |
                                                    Foundation.Core.Operation |
                                                       Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |
                                                    Foundation.Core.Parameter |
                                                    Foundation.Core.Constraint
|
                                                    Foundation.Core.Dependency
|

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |
```

```
Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.LinkEnd |

Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance |

Behavioral_Elements.Common_Behavior.AttributeLink |

Behavioral_Elements.Common_Behavior.MessageInstance |

Behavioral_Elements.Collaborations.Interaction |

Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel |
```

```
Behavioral_Elements.State_Machines.Guard |

Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)*
                                                                    >

<!ELEMENT Foundation.Core.Classifier.feature (Foundation.Core.Feature |

Foundation.Core.StructuralFeature |
                                                    Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |
                                                    Foundation.Core.Operation |
                                                      Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception)*
                                                        >

<!ELEMENT Foundation.Core.BehavioralFeature.parameter

(Foundation.Core.Parameter)* >

<!ELEMENT Foundation.Core.Association.connection (Foundation.Core.AssociationEnd
|
```

```
Behavioral_Elements.Collaborations.AssociationEndRole)*
                                                              >

<!ELEMENT Foundation.Core.AssociationEnd.qualifier

(Foundation.Core.Attribute)* >

<!ELEMENT Foundation.Core.ModelElement.taggedValue

(Foundation.Extension_Mechanisms.TaggedValue)* >

<!ELEMENT Foundation.Extension_Mechanisms.Stereotype.requiredTag

(Foundation.Extension_Mechanisms.TaggedValue)* >

<!ELEMENT Foundation.Data_Types.Enumeration.literal

(Foundation.Data_Types.EnumerationLiteral)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Signal.parameter

(Foundation.Core.Parameter)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Action.actualArgument

(Behavioral_Elements.Common_Behavior.Argument)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Link.linkRole

(Behavioral_Elements.Common_Behavior.LinkEnd)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Instance.slot

(Behavioral_Elements.Common_Behavior.AttributeLink)* >

<!ELEMENT Behavioral_Elements.Collaborations.Collaboration.interaction

(Behavioral_Elements.Collaborations.Interaction)* >

<!ELEMENT Behavioral_Elements.State_Machines.StateMachine.top
(Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)
                                                                  >
```

```
<!ELEMENT Behavioral_Elements.State_Machines.StateMachine.transitions

(Behavioral_Elements.State_Machines.Transition)* >

<!ELEMENT Behavioral_Elements.State_Machines.Transition.guard

(Behavioral_Elements.State_Machines.Guard) >

<!ELEMENT Behavioral_Elements.State_Machines.State.internalTransition

(Behavioral_Elements.State_Machines.Transition)* >

<!ELEMENT Behavioral_Elements.State_Machines.CompositeState.substate
(Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)+
                                                                    >

<!ELEMENT Behavioral_Elements.State_Machines.Transition.effect

(Behavioral_Elements.Common_Behavior.ActionSequence) >

<!ELEMENT Behavioral_Elements.State_Machines.State.entry

(Behavioral_Elements.Common_Behavior.ActionSequence) >

<!ELEMENT Behavioral_Elements.State_Machines.State.exit

(Behavioral_Elements.Common_Behavior.ActionSequence) >

<!ELEMENT Behavioral_Elements.Common_Behavior.ActionSequence.action
(Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |
```

```
Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction)*
                                                                    >

<!ELEMENT Behavioral_Elements.State_Machines.ActivityModel.partition

(Behavioral_Elements.State_Machines.Partition)* >

<!ELEMENT Foundation.Auxiliary_Elements.Binding.argument
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |
```

```
Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |
```

```
Behavioral_Elements.Common_Behavior.CallAction |
Behavioral_Elements.Common_Behavior.LocalInvocation |
Behavioral_Elements.Common_Behavior.ReturnAction |
Behavioral_Elements.Common_Behavior.SendAction |
Behavioral_Elements.Common_Behavior.UninterpretedAction |
Behavioral_Elements.Common_Behavior.TerminateAction |
Behavioral_Elements.Common_Behavior.DestroyAction |
Behavioral_Elements.Common_Behavior.Link |
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
```

```
Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)+
                                                                    >

<!ELEMENT Foundation.Core.Dependency.subDependencies (Foundation.Core.Dependency
|

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding)*
                                                          >

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL PACKAGE:  Foundation                                 -->
<!-- ____________________________________________________________ -->


<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL PACKAGE:  Auxiliary_Elements                      -->
<!-- ____________________________________________________________ -->


<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:  Refinement                                   -->
<!-- ____________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Refinement.mapping

(Foundation.Data_Types.Mapping) >
```

```
<!ELEMENT Foundation.Auxiliary_Elements.Refinement
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.Dependency.description,

Foundation.Auxiliary_Elements.Refinement.mapping,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.Dependency.client*,

Foundation.Core.Dependency.supplier*,

Foundation.Core.Dependency.owningDependency?,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Dependency.subDependencies*)?
                                                      >
<!ATTLIST Foundation.Auxiliary_Elements.Refinement
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
```

```
<!-- METAMODEL CLASS:  Usage                                                    -->
<!-- _____________________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Usage
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.Dependency.description,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.Dependency.client*,

Foundation.Core.Dependency.supplier*,

Foundation.Core.Dependency.owningDependency?,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Dependency.subDependencies*)?
                                                                    >
<!ATTLIST Foundation.Auxiliary_Elements.Usage
              %XMI.element.att;
              %XMI.link.att;
>
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Trace                                             -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Trace
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.Dependency.description,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.Dependency.client*,

Foundation.Core.Dependency.supplier*,

Foundation.Core.Dependency.owningDependency?,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Dependency.subDependencies*)?
                                                  >
<!ATTLIST Foundation.Auxiliary_Elements.Trace
              %XMI.element.att;
              %XMI.link.att;
```

```
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:  Binding                                            -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Binding
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.Dependency.description,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.Dependency.client*,

Foundation.Core.Dependency.supplier*,

Foundation.Core.Dependency.owningDependency?,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Dependency.subDependencies*,

Foundation.Auxiliary_Elements.Binding.argument+)?
```

```
                                                                    >
<!ATTLIST Foundation.Auxiliary_Elements.Binding
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Node                                              -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Node.component

(Foundation.Auxiliary_Elements.Component)* >

<!ELEMENT Foundation.Auxiliary_Elements.Node (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
```

```
Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Foundation.Auxiliary_Elements.Node.component*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*)?
                                                    >
<!ATTLIST Foundation.Auxiliary_Elements.Node
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Component                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Component.deployment

(Foundation.Auxiliary_Elements.Node)* >

<!ELEMENT Foundation.Auxiliary_Elements.Component.implements
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |
```

```
Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |
```

```
Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject |
```

```
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
Behavioral_Elements.State_Machines.ObjectFlowState |
Behavioral_Elements.State_Machines.SubmachineState |
Behavioral_Elements.State_Machines.Transition |
Behavioral_Elements.State_Machines.Event |
Behavioral_Elements.State_Machines.SignalEvent |
Behavioral_Elements.State_Machines.CallEvent |
Behavioral_Elements.State_Machines.TimeEvent |
Behavioral_Elements.State_Machines.ChangeEvent |
Foundation.Auxiliary_Elements.Comment)*
                                                    >
```

```
<!ELEMENT Foundation.Auxiliary_Elements.Component
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,
```

```
Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Foundation.Auxiliary_Elements.Component.deployment*,

Foundation.Auxiliary_Elements.Component.implements*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*)?
                                                        >
<!ATTLIST Foundation.Auxiliary_Elements.Component
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Comment                                           -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Comment
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,
```

```
Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.ModelElement.taggedValue*)?
                                                    >
<!ATTLIST Foundation.Auxiliary_Elements.Comment
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL CLASS:   ViewElement                               -->
<!-- ____________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.ViewElement.presentation

(Foundation.Auxiliary_Elements.Presentation)* >

<!ELEMENT Foundation.Auxiliary_Elements.ViewElement.model
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |
```

```
Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |
```

```
Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.LinkEnd |

Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance |

Behavioral_Elements.Common_Behavior.AttributeLink |

Behavioral_Elements.Common_Behavior.MessageInstance |

Behavioral_Elements.Collaborations.Interaction |

Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel |
```

```
Behavioral_Elements.State_Machines.Guard |

Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)*
                                                                    >

<!ELEMENT Foundation.Auxiliary_Elements.ViewElement (XMI.extension*,

Foundation.Auxiliary_Elements.ViewElement.presentation*,

Foundation.Auxiliary_Elements.ViewElement.model*)?
                                                          >
<!ATTLIST Foundation.Auxiliary_Elements.ViewElement
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   Presentation                                        -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Auxiliary_Elements.Presentation.geometry
```

```
(Foundation.Data_Types.Geometry) >

<!ELEMENT Foundation.Auxiliary_Elements.Presentation.sytle

(Foundation.Data_Types.GraphicMarker) >

<!ELEMENT Foundation.Auxiliary_Elements.Presentation.viewElement

(Foundation.Auxiliary_Elements.ViewElement)* >

<!ELEMENT Foundation.Auxiliary_Elements.Presentation.model
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |
```

```
Model_Management.Model |
Behavioral_Elements.Common_Behavior.Signal |
Behavioral_Elements.Common_Behavior.Exception |
Behavioral_Elements.Collaborations.Collaboration |
Foundation.Core.Feature |
Foundation.Core.StructuralFeature |
Foundation.Core.Attribute |
Foundation.Core.BehavioralFeature |
Foundation.Core.Operation |
Foundation.Core.Method |
Behavioral_Elements.Common_Behavior.Reception |
Foundation.Core.Parameter |
Foundation.Core.Constraint |
Foundation.Core.Dependency |
Foundation.Auxiliary_Elements.Refinement |
Foundation.Auxiliary_Elements.Usage |
Foundation.Auxiliary_Elements.Trace |
Foundation.Auxiliary_Elements.Binding |
Foundation.Core.Generalization |
Foundation.Core.AssociationEnd |
Behavioral_Elements.Collaborations.AssociationEndRole |
Behavioral_Elements.Common_Behavior.Request |
Behavioral_Elements.Common_Behavior.ActionSequence |
Behavioral_Elements.Common_Behavior.Action |
Behavioral_Elements.Common_Behavior.CreateAction |
Behavioral_Elements.Common_Behavior.CallAction |
Behavioral_Elements.Common_Behavior.LocalInvocation |
```

```
Behavioral_Elements.Common_Behavior.ReturnAction |
Behavioral_Elements.Common_Behavior.SendAction |
Behavioral_Elements.Common_Behavior.UninterpretedAction |
Behavioral_Elements.Common_Behavior.TerminateAction |
Behavioral_Elements.Common_Behavior.DestroyAction |
Behavioral_Elements.Common_Behavior.Link |
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
Behavioral_Elements.State_Machines.ObjectFlowState |
Behavioral_Elements.State_Machines.SubmachineState |
```

```
Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)*
                                                                    >

<!ELEMENT Foundation.Auxiliary_Elements.Presentation
(Foundation.Auxiliary_Elements.Presentation.geometry,

Foundation.Auxiliary_Elements.Presentation.sytle,
                                                          XMI.extension*,

Foundation.Auxiliary_Elements.Presentation.viewElement*,

Foundation.Auxiliary_Elements.Presentation.model*)?
                                                           >
<!ATTLIST Foundation.Auxiliary_Elements.Presentation
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Foundation.Auxiliary_Elements
((Foundation.Auxiliary_Elements.Refinement |
                                          Foundation.Auxiliary_Elements.Usage |
                                          Foundation.Auxiliary_Elements.Trace |
                                         Foundation.Auxiliary_Elements.Binding
|
                                           Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |
                                         Foundation.Auxiliary_Elements.Comment
|

Foundation.Auxiliary_Elements.ViewElement |

Foundation.Auxiliary_Elements.Presentation)*)
                                                  >
<!ATTLIST Foundation.Auxiliary_Elements
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL PACKAGE:  Core                                     -->
```

```
<!-- ____________________________________________________________________ -->


<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:  Element                                            -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Element (XMI.extension*)? >
<!ATTLIST Foundation.Core.Element
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:  ModelElement                                       -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.ModelElement.name (#PCDATA | XMI.reference)* >

<!ELEMENT Foundation.Core.ModelElement.visibility EMPTY >
<!ATTLIST Foundation.Core.ModelElement.visibility
              xmi.value (public | protected | private) #REQUIRED
>

<!ELEMENT Foundation.Core.ModelElement.constraint

(Foundation.Core.Constraint)* >

<!ELEMENT Foundation.Core.ModelElement.requirement (Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding)*
                                                                  >

<!ELEMENT Foundation.Core.ModelElement.provision (Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding)*
                                                                >

<!ELEMENT Foundation.Core.ModelElement.stereotype
```

```
(Foundation.Extension_Mechanisms.Stereotype)* >

<!ELEMENT Foundation.Core.ModelElement.elementReference

(Model_Management.ElementReference)* >

<!ELEMENT Foundation.Core.ModelElement.collaboration

(Behavioral_Elements.Collaborations.Collaboration)* >

<!ELEMENT Foundation.Core.ModelElement.partition

(Behavioral_Elements.State_Machines.Partition) >

<!ELEMENT Foundation.Core.ModelElement.template (Foundation.Core.ModelElement |
                                                 Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |
                                                Foundation.Core.Classifier |
                                                 Foundation.Core.Interface |
                                                     Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                   Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
                                                Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |
                                                Foundation.Core.Association
|

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |
                                                  Model_Management.Package |
                                                    Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |
```

```
Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |
                                                    Foundation.Core.Feature |

Foundation.Core.StructuralFeature |
                                                  Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |
                                                  Foundation.Core.Operation |
                                                     Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |
                                                  Foundation.Core.Parameter |
                                                 Foundation.Core.Constraint |
                                                 Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |
```

```
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
Behavioral_Elements.State_Machines.ObjectFlowState |
Behavioral_Elements.State_Machines.SubmachineState |
Behavioral_Elements.State_Machines.Transition |
Behavioral_Elements.State_Machines.Event |
Behavioral_Elements.State_Machines.SignalEvent |
Behavioral_Elements.State_Machines.CallEvent |
Behavioral_Elements.State_Machines.TimeEvent |
Behavioral_Elements.State_Machines.ChangeEvent |
```

```
Foundation.Auxiliary_Elements.Comment)
                                                                >

<!ELEMENT Foundation.Core.ModelElement.view

(Foundation.Auxiliary_Elements.ViewElement)* >

<!ELEMENT Foundation.Core.ModelElement.presentation

(Foundation.Auxiliary_Elements.Presentation)* >

<!ELEMENT Foundation.Core.ModelElement.namespace (Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |
                                                    Foundation.Core.Classifier
|
                                                    Foundation.Core.Interface |
                                                        Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                     Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
                                                    Model_Management.Subsystem
|

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |
                                                  Foundation.Core.Association
|

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |
                                                     Model_Management.Package |
                                                       Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |
```

```
Behavioral_Elements.Collaborations.Collaboration)
                                                                >

<!ELEMENT Foundation.Core.ModelElement.behavior
(Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel)*
                                                              >

<!ELEMENT Foundation.Core.ModelElement.binding

(Foundation.Auxiliary_Elements.Binding) >

<!ELEMENT Foundation.Core.ModelElement.implementation

(Foundation.Auxiliary_Elements.Component)* >

<!ELEMENT Foundation.Core.ModelElement (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                 XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,
                                              Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,
                                           Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.ModelElement.taggedValue*)?
                                                    >
<!ATTLIST Foundation.Core.ModelElement
              %XMI.element.att;
              %XMI.link.att;
>
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Namespace                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Namespace (Foundation.Core.ModelElement.name,
                                     Foundation.Core.ModelElement.visibility,
                                         XMI.extension*,
                                    Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                     Foundation.Core.ModelElement.provision*,
                                    Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                     Foundation.Core.ModelElement.partition?,
                                      Foundation.Core.ModelElement.template?,
                                         Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                     Foundation.Core.ModelElement.namespace?,
                                      Foundation.Core.ModelElement.behavior*,
                                       Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.ModelElement.taggedValue*,
                                    Foundation.Core.Namespace.ownedElement*)?
                                          >
<!ATTLIST Foundation.Core.Namespace
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   GeneralizableElement                              -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.GeneralizableElement.isRoot EMPTY >
<!ATTLIST Foundation.Core.GeneralizableElement.isRoot
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Foundation.Core.GeneralizableElement.isLeaf EMPTY >
<!ATTLIST Foundation.Core.GeneralizableElement.isLeaf
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Foundation.Core.GeneralizableElement.isAbstract EMPTY >
```

```
<!ATTLIST Foundation.Core.GeneralizableElement.isAbstract
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Foundation.Core.GeneralizableElement.generalization

(Foundation.Core.Generalization)* >

<!ELEMENT Foundation.Core.GeneralizableElement.specialization

(Foundation.Core.Generalization)* >

<!ELEMENT Foundation.Core.GeneralizableElement
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
```

```
Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*)?
                                                        >
<!ATTLIST Foundation.Core.GeneralizableElement
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Classifier                                        -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Classifier.parameter
                                                  (Foundation.Core.Parameter)*
>

<!ELEMENT Foundation.Core.Classifier.structuralFeature
(Foundation.Core.StructuralFeature |

Foundation.Core.Attribute)*
                                                                  >

<!ELEMENT Foundation.Core.Classifier.specification (Foundation.Core.Classifier |
                                                    Foundation.Core.Interface
|
                                                       Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                     Foundation.Core.DataType
|

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)*
                                                            >
```

```
<!ELEMENT Foundation.Core.Classifier.associationEnd
(Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole)*
                                                              >

<!ELEMENT Foundation.Core.Classifier.participant (Foundation.Core.AssociationEnd
|

Behavioral_Elements.Collaborations.AssociationEndRole)*
                                                          >

<!ELEMENT Foundation.Core.Classifier.createAction

(Behavioral_Elements.Common_Behavior.CreateAction)* >

<!ELEMENT Foundation.Core.Classifier.instance
(Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance)*
                                                      >

<!ELEMENT Foundation.Core.Classifier.collaboration

(Behavioral_Elements.Collaborations.Collaboration)* >

<!ELEMENT Foundation.Core.Classifier.classifierRole

(Behavioral_Elements.Collaborations.ClassifierRole)* >

<!ELEMENT Foundation.Core.Classifier.realization (Foundation.Core.Classifier |
                                                    Foundation.Core.Interface |
                                                        Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                     Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
                                                    Model_Management.Subsystem
|

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |
```

```
Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)*
                                                                >

<!ELEMENT Foundation.Core.Classifier.classifierInState

(Behavioral_Elements.State_Machines.ClassifierInState)* >

<!ELEMENT Foundation.Core.Classifier (Foundation.Core.ModelElement.name,
                                      Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                           XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                      Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                      Foundation.Core.ModelElement.partition?,
                                       Foundation.Core.ModelElement.template?,
                                           Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                      Foundation.Core.ModelElement.namespace?,
                                       Foundation.Core.ModelElement.behavior*,
                                        Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
                                         Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,
```

```
                                        Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,
                                           Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,
                                        Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Foundation.Core.ModelElement.taggedValue*,
                                        Foundation.Core.Namespace.ownedElement*,
                                           Foundation.Core.Classifier.feature*)?
                                               >
<!ATTLIST Foundation.Core.Classifier
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Interface                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Interface (Foundation.Core.ModelElement.name,
                                        Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                             XMI.extension*,
                                      Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                        Foundation.Core.ModelElement.provision*,
                                      Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                        Foundation.Core.ModelElement.partition?,
                                         Foundation.Core.ModelElement.template?,
                                            Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                        Foundation.Core.ModelElement.namespace?,
                                         Foundation.Core.ModelElement.behavior*,
                                          Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
```

```
Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
                                        Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,
                                       Foundation.Core.Classifier.participant*,
                                     Foundation.Core.Classifier.createAction*,
                                         Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,
                                       Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Foundation.Core.ModelElement.taggedValue*,
                                       Foundation.Core.Namespace.ownedElement*,
                                          Foundation.Core.Classifier.feature*)?
                                            >
<!ATTLIST Foundation.Core.Interface
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:  Class                                              -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Class.isActive EMPTY >
<!ATTLIST Foundation.Core.Class.isActive
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Foundation.Core.Class (Foundation.Core.ModelElement.name,
                                       Foundation.Core.ModelElement.visibility,
                                  Foundation.Core.GeneralizableElement.isRoot,
                                  Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                       Foundation.Core.Class.isActive,
                                       XMI.extension*,
                                      Foundation.Core.ModelElement.constraint*,
                                    Foundation.Core.ModelElement.requirement*,
                                       Foundation.Core.ModelElement.provision*,
                                      Foundation.Core.ModelElement.stereotype*,
```

```
Foundation.Core.ModelElement.elementReference*,
                                        Foundation.Core.ModelElement.collaboration*,
                                           Foundation.Core.ModelElement.partition?,
                                           Foundation.Core.ModelElement.template?,
                                           Foundation.Core.ModelElement.view*,
                                         Foundation.Core.ModelElement.presentation*,
                                           Foundation.Core.ModelElement.namespace?,
                                           Foundation.Core.ModelElement.behavior*,
                                           Foundation.Core.ModelElement.binding?,
                                       Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
                                           Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,
                                         Foundation.Core.Classifier.specification*,
                                        Foundation.Core.Classifier.associationEnd*,
                                           Foundation.Core.Classifier.participant*,
                                          Foundation.Core.Classifier.createAction*,
                                           Foundation.Core.Classifier.instance*,
                                         Foundation.Core.Classifier.collaboration*,
                                        Foundation.Core.Classifier.classifierRole*,
                                           Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,
                                         Foundation.Core.ModelElement.taggedValue*,
                                           Foundation.Core.Namespace.ownedElement*,
                                           Foundation.Core.Classifier.feature*)? >
<!ATTLIST Foundation.Core.Class
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   DataType                                          -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.DataType  (Foundation.Core.ModelElement.name,
                                        Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                             XMI.extension*,
                                          Foundation.Core.ModelElement.constraint*,
                                         Foundation.Core.ModelElement.requirement*,
                                           Foundation.Core.ModelElement.provision*,
                                          Foundation.Core.ModelElement.stereotype*,
```

```
Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                        Foundation.Core.ModelElement.partition?,
                                         Foundation.Core.ModelElement.template?,
                                           Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                        Foundation.Core.ModelElement.namespace?,
                                         Foundation.Core.ModelElement.behavior*,
                                          Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
                                            Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,
                                       Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,
                                         Foundation.Core.Classifier.participant*,
                                        Foundation.Core.Classifier.createAction*,
                                            Foundation.Core.Classifier.instance*,
                                       Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,
                                         Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,
                                       Foundation.Core.ModelElement.taggedValue*,
                                         Foundation.Core.Namespace.ownedElement*,
                                            Foundation.Core.Classifier.feature*)?
                                             >
<!ATTLIST Foundation.Core.DataType
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Feature                                           -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Feature.ownerScope EMPTY >
<!ATTLIST Foundation.Core.Feature.ownerScope
              xmi.value (classifier | instance) #REQUIRED
>

<!ELEMENT Foundation.Core.Feature.owner (Foundation.Core.Classifier |
                                                 Foundation.Core.Interface |
```

```
                                                    Foundation.Core.Class |
                                                Foundation.Core.AssociationClass |
                                                    Foundation.Core.DataType |
                                                Foundation.Data_Types.Enumeration |
                                                  Foundation.Data_Types.Primitive |
                                                  Foundation.Data_Types.Structure |
                                                    Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |
                                                Behavioral_Elements.Use_Cases.Actor |
                                               Behavioral_Elements.Use_Cases.UseCase
|

Behavioral_Elements.State_Machines.ClassifierInState |
                                                Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)
                                                      >

<!ELEMENT Foundation.Core.Feature.classifierRole

(Behavioral_Elements.Collaborations.ClassifierRole)* >

<!ELEMENT Foundation.Core.Feature (Foundation.Core.ModelElement.name,
                                          Foundation.Core.ModelElement.visibility,
                                            Foundation.Core.Feature.ownerScope,
                                            XMI.extension*,
                                         Foundation.Core.ModelElement.constraint*,
                                        Foundation.Core.ModelElement.requirement*,
                                          Foundation.Core.ModelElement.provision*,
                                         Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                          Foundation.Core.ModelElement.partition?,
                                           Foundation.Core.ModelElement.template?,
                                            Foundation.Core.ModelElement.view*,
                                       Foundation.Core.ModelElement.presentation*,
                                          Foundation.Core.ModelElement.namespace?,
                                           Foundation.Core.ModelElement.behavior*,
                                            Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                            Foundation.Core.Feature.owner?,
                                          Foundation.Core.Feature.classifierRole*,
                                       Foundation.Core.ModelElement.taggedValue*)?
                                              >
<!ATTLIST Foundation.Core.Feature
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
```

```
<!-- METAMODEL CLASS:   StructuralFeature                                    -->
<!-- ______________________________________________________________________ -->


<!ELEMENT Foundation.Core.StructuralFeature.multiplicity (#PCDATA |
                                                          XMI.reference)*
                                                               >

<!ELEMENT Foundation.Core.StructuralFeature.changeable EMPTY >
<!ATTLIST Foundation.Core.StructuralFeature.changeable
              xmi.value (none | frozen | addOnly) #REQUIRED
>

<!ELEMENT Foundation.Core.StructuralFeature.targetScope EMPTY >
<!ATTLIST Foundation.Core.StructuralFeature.targetScope
              xmi.value (classifier | instance) #REQUIRED
>

<!ELEMENT Foundation.Core.StructuralFeature.type (Foundation.Core.Classifier |
                                                    Foundation.Core.Interface |
                                                        Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                     Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
                                                    Model_Management.Subsystem
|

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)
                                                                  >

<!ELEMENT Foundation.Core.StructuralFeature (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.Feature.ownerScope,

Foundation.Core.StructuralFeature.multiplicity,

Foundation.Core.StructuralFeature.changeable,
```

```
Foundation.Core.StructuralFeature.targetScope,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                          Foundation.Core.Feature.owner?,

Foundation.Core.Feature.classifierRole*,

Foundation.Core.StructuralFeature.type,

Foundation.Core.ModelElement.taggedValue*)?
                                                  >
<!ATTLIST Foundation.Core.StructuralFeature
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   BehavioralFeature                           -->
<!-- ____________________________________________________________ -->


<!ELEMENT Foundation.Core.BehavioralFeature.isQuery EMPTY >
<!ATTLIST Foundation.Core.BehavioralFeature.isQuery
              xmi.value (false | true) #REQUIRED
>
```

```
<!ELEMENT Foundation.Core.BehavioralFeature.raisedException

(Behavioral_Elements.Common_Behavior.Exception)* >

<!ELEMENT Foundation.Core.BehavioralFeature (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.Feature.ownerScope,

Foundation.Core.BehavioralFeature.isQuery,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                                Foundation.Core.Feature.owner?,

Foundation.Core.Feature.classifierRole*,

Foundation.Core.BehavioralFeature.raisedException*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.BehavioralFeature.parameter*)?
                                                          >
<!ATTLIST Foundation.Core.BehavioralFeature
              %XMI.element.att;
              %XMI.link.att;
>
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Operation                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Operation.specification (#PCDATA |
                                                    XMI.reference)* >

<!ELEMENT Foundation.Core.Operation.isPolymorphic EMPTY >
<!ATTLIST Foundation.Core.Operation.isPolymorphic
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Foundation.Core.Operation.concurrency EMPTY >
<!ATTLIST Foundation.Core.Operation.concurrency
              xmi.value (sequential | guarded | concurrent) #REQUIRED
>

<!ELEMENT Foundation.Core.Operation.method (Foundation.Core.Method)* >

<!ELEMENT Foundation.Core.Operation.collaboration

(Behavioral_Elements.Collaborations.Collaboration)* >

<!ELEMENT Foundation.Core.Operation.occurrence

(Behavioral_Elements.State_Machines.CallEvent)* >

<!ELEMENT Foundation.Core.Operation (Foundation.Core.ModelElement.name,
                                     Foundation.Core.ModelElement.visibility,
                                         Foundation.Core.Feature.ownerScope,

Foundation.Core.BehavioralFeature.isQuery,
                                     Foundation.Core.Operation.specification,
                                     Foundation.Core.Operation.isPolymorphic,
                                       Foundation.Core.Operation.concurrency,
                                         XMI.extension*,
                                    Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                     Foundation.Core.ModelElement.provision*,
                                    Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                     Foundation.Core.ModelElement.partition?,
                                      Foundation.Core.ModelElement.template?,
                                         Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                     Foundation.Core.ModelElement.namespace?,
                                      Foundation.Core.ModelElement.behavior*,
                                       Foundation.Core.ModelElement.binding?,
```

```
Foundation.Core.ModelElement.implementation*,
                                              Foundation.Core.Feature.owner?,
                                          Foundation.Core.Feature.classifierRole*,

Foundation.Core.BehavioralFeature.raisedException*,

Behavioral_Elements.Common_Behavior.Request.action*,

Behavioral_Elements.Common_Behavior.Request.messageInstance*,
                                              Foundation.Core.Operation.method*,
                                        Foundation.Core.Operation.collaboration*,
                                           Foundation.Core.Operation.occurrence*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.BehavioralFeature.parameter*)?
                                                 >
<!ATTLIST Foundation.Core.Operation
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Method                                            -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Method.body

(Foundation.Data_Types.ProcedureExpression) >

<!ELEMENT Foundation.Core.Method.specification
                                                        (Foundation.Core.Operation) >

<!ELEMENT Foundation.Core.Method (Foundation.Core.ModelElement.name,
                                        Foundation.Core.ModelElement.visibility,
                                         Foundation.Core.Feature.ownerScope,
                                     Foundation.Core.BehavioralFeature.isQuery,
                                        Foundation.Core.Method.body,
                                        XMI.extension*,
                                      Foundation.Core.ModelElement.constraint*,
                                     Foundation.Core.ModelElement.requirement*,
                                       Foundation.Core.ModelElement.provision*,
                                      Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,
                                    Foundation.Core.ModelElement.collaboration*,
                                        Foundation.Core.ModelElement.partition?,
                                         Foundation.Core.ModelElement.template?,
                                         Foundation.Core.ModelElement.view*,
                                   Foundation.Core.ModelElement.presentation*,
                                        Foundation.Core.ModelElement.namespace?,
                                         Foundation.Core.ModelElement.behavior*,
```

```
                                          Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                          Foundation.Core.Feature.owner?,
                                         Foundation.Core.Feature.classifierRole*,

Foundation.Core.BehavioralFeature.raisedException*,
                                          Foundation.Core.Method.specification,
                                       Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.BehavioralFeature.parameter*)?
                                           >
<!ATTLIST Foundation.Core.Method
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Parameter                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Parameter.defaultValue

(Foundation.Data_Types.Expression) >

<!ELEMENT Foundation.Core.Parameter.kind EMPTY >
<!ATTLIST Foundation.Core.Parameter.kind
              xmi.value (in | inout | out | return) #REQUIRED
>

<!ELEMENT Foundation.Core.Parameter.type (Foundation.Core.Classifier |
                                              Foundation.Core.Interface |
                                              Foundation.Core.Class |
                                         Foundation.Core.AssociationClass |
                                              Foundation.Core.DataType |
                                        Foundation.Data_Types.Enumeration |
                                          Foundation.Data_Types.Primitive |
                                          Foundation.Data_Types.Structure |
                                              Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |
                                         Behavioral_Elements.Use_Cases.Actor
|

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |
                                         Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)
                                                 >
```

```
<!ELEMENT Foundation.Core.Parameter.behavioralFeature
(Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |
                                                    Foundation.Core.Method
|

Behavioral_Elements.Common_Behavior.Reception)
                                                                    >

<!ELEMENT Foundation.Core.Parameter.signal
(Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception)
                                                  >

<!ELEMENT Foundation.Core.Parameter (Foundation.Core.ModelElement.name,
                                     Foundation.Core.ModelElement.visibility,
                                      Foundation.Core.Parameter.defaultValue,
                                         Foundation.Core.Parameter.kind,
                                         XMI.extension*,
                                    Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                     Foundation.Core.ModelElement.provision*,
                                    Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                     Foundation.Core.ModelElement.partition?,
                                      Foundation.Core.ModelElement.template?,
                                         Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                     Foundation.Core.ModelElement.namespace?,
                                      Foundation.Core.ModelElement.behavior*,
                                       Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                         Foundation.Core.Parameter.type,

Foundation.Core.Parameter.behavioralFeature?,
                                         Foundation.Core.Parameter.signal?,

Foundation.Core.ModelElement.taggedValue*)?
                                           >
<!ATTLIST Foundation.Core.Parameter
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Constraint                                        -->
```

```
<!-- ____________________________________________________________ -->

<!ELEMENT Foundation.Core.Constraint.body

(Foundation.Data_Types.BooleanExpression) >

<!ELEMENT Foundation.Core.Constraint.constrainedElement
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |
```

```
Behavioral_Elements.Common_Behavior.Exception |
Behavioral_Elements.Collaborations.Collaboration |
Foundation.Core.Feature |
Foundation.Core.StructuralFeature |
Foundation.Core.Attribute |
Foundation.Core.BehavioralFeature |
Foundation.Core.Operation |
Foundation.Core.Method |
Behavioral_Elements.Common_Behavior.Reception |
Foundation.Core.Parameter |
Foundation.Core.Constraint |
Foundation.Core.Dependency |
Foundation.Auxiliary_Elements.Refinement |
Foundation.Auxiliary_Elements.Usage |
Foundation.Auxiliary_Elements.Trace |
Foundation.Auxiliary_Elements.Binding |
Foundation.Core.Generalization |
Foundation.Core.AssociationEnd |
Behavioral_Elements.Collaborations.AssociationEndRole |
Behavioral_Elements.Common_Behavior.Request |
Behavioral_Elements.Common_Behavior.ActionSequence |
Behavioral_Elements.Common_Behavior.Action |
Behavioral_Elements.Common_Behavior.CreateAction |
Behavioral_Elements.Common_Behavior.CallAction |
Behavioral_Elements.Common_Behavior.LocalInvocation |
Behavioral_Elements.Common_Behavior.ReturnAction |
Behavioral_Elements.Common_Behavior.SendAction |
```

```
Behavioral_Elements.Common_Behavior.UninterpretedAction |
Behavioral_Elements.Common_Behavior.TerminateAction |
Behavioral_Elements.Common_Behavior.DestroyAction |
Behavioral_Elements.Common_Behavior.Link |
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
Behavioral_Elements.State_Machines.ObjectFlowState |
Behavioral_Elements.State_Machines.SubmachineState |
Behavioral_Elements.State_Machines.Transition |
Behavioral_Elements.State_Machines.Event |
```

```
Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)+
                                                                    >

<!ELEMENT Foundation.Core.Constraint.constrainedStereotype

(Foundation.Extension_Mechanisms.Stereotype) >

<!ELEMENT Foundation.Core.Constraint (Foundation.Core.ModelElement.name,
                                      Foundation.Core.ModelElement.visibility,
                                           Foundation.Core.Constraint.body,
                                           XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                      Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                      Foundation.Core.ModelElement.partition?,
                                       Foundation.Core.ModelElement.template?,
                                           Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                      Foundation.Core.ModelElement.namespace?,
                                       Foundation.Core.ModelElement.behavior*,
                                        Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.Constraint.constrainedElement+,

Foundation.Core.Constraint.constrainedStereotype?,

Foundation.Core.ModelElement.taggedValue*)?
                                                 >
<!ATTLIST Foundation.Core.Constraint
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
```

```
<!-- METAMODEL CLASS:   Dependency                                          -->
<!-- ______________________________________________________________ -->


<!ELEMENT Foundation.Core.Dependency.description (#PCDATA |
                                                  XMI.reference)* >

<!ELEMENT Foundation.Core.Dependency.client (Foundation.Core.ModelElement |
                                             Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |
                                           Foundation.Core.Classifier |
                                            Foundation.Core.Interface |
                                              Foundation.Core.Class |
                                        Foundation.Core.AssociationClass
|
                                               Foundation.Core.DataType |
                                     Foundation.Data_Types.Enumeration
|
                                       Foundation.Data_Types.Primitive
|
                                       Foundation.Data_Types.Structure
|
                                           Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |
                                         Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |
                                            Model_Management.Package |
                                            Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |
                                              Foundation.Core.Feature |
                                     Foundation.Core.StructuralFeature
|
                                            Foundation.Core.Attribute |
                                     Foundation.Core.BehavioralFeature
|
```

```
                                                        Foundation.Core.Operation |
                                                         Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |
                                                        Foundation.Core.Parameter |
                                                      Foundation.Core.Constraint |
                                                      Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |
                                                  Foundation.Core.Generalization |
                                                  Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.LinkEnd |

Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance |
```

```
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
Behavioral_Elements.State_Machines.ObjectFlowState |
Behavioral_Elements.State_Machines.SubmachineState |
Behavioral_Elements.State_Machines.Transition |
Behavioral_Elements.State_Machines.Event |
Behavioral_Elements.State_Machines.SignalEvent |
Behavioral_Elements.State_Machines.CallEvent |
Behavioral_Elements.State_Machines.TimeEvent |
Behavioral_Elements.State_Machines.ChangeEvent |
Foundation.Auxiliary_Elements.Comment)*
                                                        >

<!ELEMENT Foundation.Core.Dependency.supplier (Foundation.Core.ModelElement |
                                                  Foundation.Core.Namespace |
Foundation.Core.GeneralizableElement |
                                                Foundation.Core.Classifier |
                                                 Foundation.Core.Interface |
                                                    Foundation.Core.Class |

Foundation.Core.AssociationClass |
```

```
                                                    Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
                                                  Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |
                                                Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |
                                                    Model_Management.Package |
                                                      Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |
                                                     Foundation.Core.Feature |

Foundation.Core.StructuralFeature |
                                                   Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |
                                                   Foundation.Core.Operation |
                                                      Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |
                                                   Foundation.Core.Parameter |
                                                  Foundation.Core.Constraint |
                                                  Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |
```

```
                                                    Foundation.Core.Generalization
|
                                                    Foundation.Core.AssociationEnd
|
Behavioral_Elements.Collaborations.AssociationEndRole |
Behavioral_Elements.Common_Behavior.Request |
Behavioral_Elements.Common_Behavior.ActionSequence |
Behavioral_Elements.Common_Behavior.Action |
Behavioral_Elements.Common_Behavior.CreateAction |
Behavioral_Elements.Common_Behavior.CallAction |
Behavioral_Elements.Common_Behavior.LocalInvocation |
Behavioral_Elements.Common_Behavior.ReturnAction |
Behavioral_Elements.Common_Behavior.SendAction |
Behavioral_Elements.Common_Behavior.UninterpretedAction |
Behavioral_Elements.Common_Behavior.TerminateAction |
Behavioral_Elements.Common_Behavior.DestroyAction |
Behavioral_Elements.Common_Behavior.Link |
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
```

```
Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)*
                                                        >

<!ELEMENT Foundation.Core.Dependency.owningDependency
(Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding)
                                                                  >

<!ELEMENT Foundation.Core.Dependency (Foundation.Core.ModelElement.name,
                                      Foundation.Core.ModelElement.visibility,
                                       Foundation.Core.Dependency.description,
                                           XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                      Foundation.Core.ModelElement.provision*,
```

```
Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                        Foundation.Core.ModelElement.partition?,
                                         Foundation.Core.ModelElement.template?,
                                             Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                        Foundation.Core.ModelElement.namespace?,
                                         Foundation.Core.ModelElement.behavior*,
                                          Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                            Foundation.Core.Dependency.client*,
                                           Foundation.Core.Dependency.supplier*,

Foundation.Core.Dependency.owningDependency?,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Dependency.subDependencies*)?
                                                 >
<!ATTLIST Foundation.Core.Dependency
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Generalization                                    -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Generalization.discriminator (#PCDATA |
                                                              XMI.reference)* >

<!ELEMENT Foundation.Core.Generalization.subtype
(Foundation.Core.GeneralizableElement |
                                                    Foundation.Core.Classifier
|
                                                    Foundation.Core.Interface |
                                                        Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                     Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
```

```
                                                    Model_Management.Subsystem
|

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |
                                                  Foundation.Core.Association
|

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |
                                                    Model_Management.Package |
                                                      Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception)
                                                                        >

<!ELEMENT Foundation.Core.Generalization.supertype
(Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |
                                                   Foundation.Core.Interface
|
                                                      Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                    Foundation.Core.DataType
|

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |
```

```
Foundation.Auxiliary_Elements.Node |
Foundation.Auxiliary_Elements.Component |
Foundation.Core.Association |
Behavioral_Elements.Collaborations.AssociationRole |
Foundation.Extension_Mechanisms.Stereotype |
                                                    Model_Management.Package
|
                                                     Model_Management.Model |
Behavioral_Elements.Common_Behavior.Signal |
Behavioral_Elements.Common_Behavior.Exception)
                                                                          >
<!ELEMENT Foundation.Core.Generalization (Foundation.Core.ModelElement.name,
Foundation.Core.ModelElement.visibility,
Foundation.Core.Generalization.discriminator,
                                                  XMI.extension*,
Foundation.Core.ModelElement.constraint*,
Foundation.Core.ModelElement.requirement*,
Foundation.Core.ModelElement.provision*,
Foundation.Core.ModelElement.stereotype*,
Foundation.Core.ModelElement.elementReference*,
Foundation.Core.ModelElement.collaboration*,
Foundation.Core.ModelElement.partition?,
Foundation.Core.ModelElement.template?,
                                              Foundation.Core.ModelElement.view*,
Foundation.Core.ModelElement.presentation*,
Foundation.Core.ModelElement.namespace?,
Foundation.Core.ModelElement.behavior*,
Foundation.Core.ModelElement.binding?,
Foundation.Core.ModelElement.implementation*,
Foundation.Core.Generalization.subtype,
```

```
Foundation.Core.Generalization.supertype,

Foundation.Core.ModelElement.taggedValue*)?
                                                    >
<!ATTLIST Foundation.Core.Generalization
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   AssociationEnd                                    -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.AssociationEnd.isNavigable EMPTY >
<!ATTLIST Foundation.Core.AssociationEnd.isNavigable
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Foundation.Core.AssociationEnd.isOrdered EMPTY >
<!ATTLIST Foundation.Core.AssociationEnd.isOrdered
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Foundation.Core.AssociationEnd.aggregation EMPTY >
<!ATTLIST Foundation.Core.AssociationEnd.aggregation
              xmi.value (none | shared | composite) #REQUIRED
>

<!ELEMENT Foundation.Core.AssociationEnd.multiplicity (#PCDATA |
                                                       XMI.reference)* >

<!ELEMENT Foundation.Core.AssociationEnd.changeable EMPTY >
<!ATTLIST Foundation.Core.AssociationEnd.changeable
              xmi.value (none | frozen | addOnly) #REQUIRED
>

<!ELEMENT Foundation.Core.AssociationEnd.targetScope EMPTY >
<!ATTLIST Foundation.Core.AssociationEnd.targetScope
              xmi.value (classifier | instance) #REQUIRED
>

<!ELEMENT Foundation.Core.AssociationEnd.type (Foundation.Core.Classifier |
                                                  Foundation.Core.Interface |
                                                     Foundation.Core.Class |

Foundation.Core.AssociationClass |
                                                   Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |
```

```
Foundation.Data_Types.Structure |
                                                    Model_Management.Subsystem |
Behavioral_Elements.Collaborations.ClassifierRole |
Behavioral_Elements.Use_Cases.Actor |
Behavioral_Elements.Use_Cases.UseCase |
Behavioral_Elements.State_Machines.ClassifierInState |
Foundation.Auxiliary_Elements.Node |
Foundation.Auxiliary_Elements.Component)
                                                  >

<!ELEMENT Foundation.Core.AssociationEnd.specification
(Foundation.Core.Classifier |
Foundation.Core.Interface |
                                                    Foundation.Core.Class
|
Foundation.Core.AssociationClass |
Foundation.Core.DataType |
Foundation.Data_Types.Enumeration |
Foundation.Data_Types.Primitive |
Foundation.Data_Types.Structure |
Model_Management.Subsystem |
Behavioral_Elements.Collaborations.ClassifierRole |
Behavioral_Elements.Use_Cases.Actor |
Behavioral_Elements.Use_Cases.UseCase |
Behavioral_Elements.State_Machines.ClassifierInState |
Foundation.Auxiliary_Elements.Node |
Foundation.Auxiliary_Elements.Component)*
                                                                  >

<!ELEMENT Foundation.Core.AssociationEnd.linkEnd
(Behavioral_Elements.Common_Behavior.LinkEnd)* >
<!ELEMENT Foundation.Core.AssociationEnd.associationEndRole
```

```
(Behavioral_Elements.Collaborations.AssociationEndRole)* >

<!ELEMENT Foundation.Core.AssociationEnd.association
(Foundation.Core.Association |

Foundation.Core.AssociationClass |

Behavioral_Elements.Collaborations.AssociationRole)
                                                                    >

<!ELEMENT Foundation.Core.AssociationEnd (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.AssociationEnd.isNavigable,

Foundation.Core.AssociationEnd.isOrdered,

Foundation.Core.AssociationEnd.aggregation,

Foundation.Core.AssociationEnd.multiplicity,

Foundation.Core.AssociationEnd.changeable,

Foundation.Core.AssociationEnd.targetScope,
                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,
                                            Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                           Foundation.Core.AssociationEnd.type,
```

```
Foundation.Core.AssociationEnd.specification*,

Foundation.Core.AssociationEnd.linkEnd*,

Foundation.Core.AssociationEnd.associationEndRole*,

Foundation.Core.AssociationEnd.association?,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.AssociationEnd.qualifier*)?
                                                    >
<!ATTLIST Foundation.Core.AssociationEnd
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Association                                       -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Association.link
(Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject)*
                                                      >

<!ELEMENT Foundation.Core.Association.associationEnd

(Behavioral_Elements.Collaborations.AssociationRole)* >

<!ELEMENT Foundation.Core.Association (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,
```

```
Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,
                                        Foundation.Core.ModelElement.template?,
                                            Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,
                                        Foundation.Core.ModelElement.behavior*,
                                         Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
                                                Foundation.Core.Association.link*,

Foundation.Core.Association.associationEnd*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Association.connection*)?
                                                    >
<!ATTLIST Foundation.Core.Association
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   AssociationClass                                  -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.AssociationClass (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                  Foundation.Core.Class.isActive,
                                                      XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
```

```
Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,
                                                Foundation.Core.Association.link*,

Foundation.Core.Association.associationEnd*,

Foundation.Core.ModelElement.taggedValue*,
```

```
Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*,

Foundation.Core.Association.connection*)?
                                                        >
<!ATTLIST Foundation.Core.AssociationClass
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Attribute                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Core.Attribute.initialValue

(Foundation.Data_Types.Expression) >

<!ELEMENT Foundation.Core.Attribute.attributeLink

(Behavioral_Elements.Common_Behavior.AttributeLink)* >

<!ELEMENT Foundation.Core.Attribute.associationEnd
(Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole)
                                                              >

<!ELEMENT Foundation.Core.Attribute (Foundation.Core.ModelElement.name,
                                        Foundation.Core.ModelElement.visibility,
                                            Foundation.Core.Feature.ownerScope,

Foundation.Core.StructuralFeature.multiplicity,

Foundation.Core.StructuralFeature.changeable,

Foundation.Core.StructuralFeature.targetScope,
                                             Foundation.Core.Attribute.initialValue,
                                                XMI.extension*,
                                           Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                             Foundation.Core.ModelElement.provision*,
                                           Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                           Foundation.Core.ModelElement.partition?,
                                            Foundation.Core.ModelElement.template?,
                                               Foundation.Core.ModelElement.view*,
```

```
Foundation.Core.ModelElement.presentation*,
                                        Foundation.Core.ModelElement.namespace?,
                                         Foundation.Core.ModelElement.behavior*,
                                          Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
                                            Foundation.Core.Feature.owner?,
                                        Foundation.Core.Feature.classifierRole*,
                                         Foundation.Core.StructuralFeature.type,
                                       Foundation.Core.Attribute.attributeLink*,

Foundation.Core.Attribute.associationEnd?,

Foundation.Core.ModelElement.taggedValue*)?
                                                 >
<!ATTLIST Foundation.Core.Attribute
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Foundation.Core ((Foundation.Core.Element |
                                Foundation.Core.ModelElement |
                                Foundation.Core.Namespace |
                                Foundation.Core.GeneralizableElement |
                                Foundation.Core.Classifier |
                                Foundation.Core.Interface |
                                Foundation.Core.Class |
                                Foundation.Core.DataType |
                                Foundation.Core.Feature |
                                Foundation.Core.StructuralFeature |
                                Foundation.Core.BehavioralFeature |
                                Foundation.Core.Operation |
                                Foundation.Core.Method |
                                Foundation.Core.Parameter |
                                Foundation.Core.Constraint |
                                Foundation.Core.Dependency |
                                Foundation.Core.Generalization |
                                Foundation.Core.AssociationEnd |
                                Foundation.Core.Association |
                                Foundation.Core.AssociationClass |
                                Foundation.Core.Attribute)*) >
<!ATTLIST Foundation.Core
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL PACKAGE:   Extension_Mechanisms                            -->
<!-- ____________________________________________________________________ -->


<!-- ____________________________________________________________________ -->
<!--                                                                      -->
```

```
<!-- METAMODEL CLASS:   TaggedValue                                          -->
<!-- ______________________________________________________________________ -->


<!ELEMENT Foundation.Extension_Mechanisms.TaggedValue.tag (#PCDATA |
                                                          XMI.reference)*
                                                                 >

<!ELEMENT Foundation.Extension_Mechanisms.TaggedValue.value (#PCDATA |
                                                          XMI.reference)*
                                                                   >

<!ELEMENT Foundation.Extension_Mechanisms.TaggedValue.modelElement
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |
```

```
Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |
```

```
Behavioral_Elements.Common_Behavior.LocalInvocation |
Behavioral_Elements.Common_Behavior.ReturnAction |
Behavioral_Elements.Common_Behavior.SendAction |
Behavioral_Elements.Common_Behavior.UninterpretedAction |
Behavioral_Elements.Common_Behavior.TerminateAction |
Behavioral_Elements.Common_Behavior.DestroyAction |
Behavioral_Elements.Common_Behavior.Link |
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
Behavioral_Elements.State_Machines.ObjectFlowState |
```

```
Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)
                                                                        >

<!ELEMENT Foundation.Extension_Mechanisms.TaggedValue.stereotype

(Foundation.Extension_Mechanisms.Stereotype) >

<!ELEMENT Foundation.Extension_Mechanisms.TaggedValue
(Foundation.Extension_Mechanisms.TaggedValue.tag,

Foundation.Extension_Mechanisms.TaggedValue.value,
                                                    XMI.extension*,

Foundation.Extension_Mechanisms.TaggedValue.modelElement?,

Foundation.Extension_Mechanisms.TaggedValue.stereotype?)?
                                                      >
<!ATTLIST Foundation.Extension_Mechanisms.TaggedValue
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   Stereotype                                  -->
<!-- ____________________________________________________________ -->


<!ELEMENT Foundation.Extension_Mechanisms.Stereotype.icon

(Foundation.Data_Types.Geometry) >

<!ELEMENT Foundation.Extension_Mechanisms.Stereotype.baseClass (#PCDATA |

XMI.reference)*
                                                                        >

<!ELEMENT Foundation.Extension_Mechanisms.Stereotype.extendedElement
(Foundation.Core.ModelElement |
```

```
Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |
```

```
Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |
```

```
Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.LinkEnd |

Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance |

Behavioral_Elements.Common_Behavior.AttributeLink |

Behavioral_Elements.Common_Behavior.MessageInstance |

Behavioral_Elements.Collaborations.Interaction |

Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel |

Behavioral_Elements.State_Machines.Guard |

Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |
```

```
Foundation.Auxiliary_Elements.Comment)*
                                                                                >

<!ELEMENT Foundation.Extension_Mechanisms.Stereotype.stereotypeConstraint

(Foundation.Core.Constraint)* >

<!ELEMENT Foundation.Extension_Mechanisms.Stereotype
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,

Foundation.Extension_Mechanisms.Stereotype.icon,

Foundation.Extension_Mechanisms.Stereotype.baseClass,
                                                          XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
```

```
Foundation.Extension_Mechanisms.Stereotype.extendedElement*,

Foundation.Extension_Mechanisms.Stereotype.stereotypeConstraint*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Extension_Mechanisms.Stereotype.requiredTag*)?
                                                              >
<!ATTLIST Foundation.Extension_Mechanisms.Stereotype
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Foundation.Extension_Mechanisms
((Foundation.Extension_Mechanisms.TaggedValue |

Foundation.Extension_Mechanisms.Stereotype)*)
                                                    >
<!ATTLIST Foundation.Extension_Mechanisms
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL PACKAGE:   Data_Types                                        -->
<!-- ____________________________________________________________________ -->


<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   Enumeration                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.Enumeration (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,
```

```
Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*,
```

```
Foundation.Data_Types.Enumeration.literal*)?
                                                        >
<!ATTLIST Foundation.Data_Types.Enumeration
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   EnumerationLiteral                                -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.EnumerationLiteral.name (#PCDATA |
                                                        XMI.reference)* >

<!ELEMENT Foundation.Data_Types.EnumerationLiteral.enumeration

(Foundation.Data_Types.Enumeration) >

<!ELEMENT Foundation.Data_Types.EnumerationLiteral
(Foundation.Data_Types.EnumerationLiteral.name,
                                                    XMI.extension*,

Foundation.Data_Types.EnumerationLiteral.enumeration?)?
                                                     >
<!ATTLIST Foundation.Data_Types.EnumerationLiteral
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Primitive                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.Primitive (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,
```

```
Foundation.Core.ModelElement.stereotype*,
Foundation.Core.ModelElement.elementReference*,
Foundation.Core.ModelElement.collaboration*,
Foundation.Core.ModelElement.partition?,
Foundation.Core.ModelElement.template?,
                                        Foundation.Core.ModelElement.view*,
Foundation.Core.ModelElement.presentation*,
Foundation.Core.ModelElement.namespace?,
Foundation.Core.ModelElement.behavior*,
Foundation.Core.ModelElement.binding?,
Foundation.Core.ModelElement.implementation*,
Foundation.Core.GeneralizableElement.generalization*,
Foundation.Core.GeneralizableElement.specialization*,
Foundation.Core.Classifier.parameter*,
Foundation.Core.Classifier.structuralFeature*,
Foundation.Core.Classifier.specification*,
Foundation.Core.Classifier.associationEnd*,
Foundation.Core.Classifier.participant*,
Foundation.Core.Classifier.createAction*,
Foundation.Core.Classifier.instance*,
Foundation.Core.Classifier.collaboration*,
Foundation.Core.Classifier.classifierRole*,
Foundation.Core.Classifier.realization*,
Foundation.Core.Classifier.classifierInState*,
Foundation.Core.ModelElement.taggedValue*,
Foundation.Core.Namespace.ownedElement*,
Foundation.Core.Classifier.feature*)?
                                                  >
<!ATTLIST Foundation.Data_Types.Primitive
```

```
                %XMI.element.att;
                %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Structure                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.Structure (Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,
                                              Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,
```

```
Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*)?
                                                    >
<!ATTLIST Foundation.Data_Types.Structure
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   MultiplicityRange                                 -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.MultiplicityRange.lower (#PCDATA |
                                                        XMI.reference)* >

<!ELEMENT Foundation.Data_Types.MultiplicityRange.upper (#PCDATA |
                                                        XMI.reference)* >

<!ELEMENT Foundation.Data_Types.MultiplicityRange
(Foundation.Data_Types.MultiplicityRange.lower,

Foundation.Data_Types.MultiplicityRange.upper,
                                                    XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.MultiplicityRange
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Geometry                                          -->
```

```
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.Geometry.body (#PCDATA |
                                                XMI.reference)* >

<!ELEMENT Foundation.Data_Types.Geometry (Foundation.Data_Types.Geometry.body,
                                          XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.Geometry
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   GraphicMarker                                     -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.GraphicMarker.body (#PCDATA |
                                                    XMI.reference)* >

<!ELEMENT Foundation.Data_Types.GraphicMarker
(Foundation.Data_Types.GraphicMarker.body,
                                                   XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.GraphicMarker
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Mapping                                           -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.Mapping.body (#PCDATA |
                                                  XMI.reference)* >

<!ELEMENT Foundation.Data_Types.Mapping (Foundation.Data_Types.Mapping.body,
                                         XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.Mapping
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Expression                                        -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.Expression.language (#PCDATA |
                                                     XMI.reference)* >
```

```
<!ELEMENT Foundation.Data_Types.Expression.body (#PCDATA |
                                                    XMI.reference)* >

<!ELEMENT Foundation.Data_Types.Expression
(Foundation.Data_Types.Expression.language,

Foundation.Data_Types.Expression.body,
                                                XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.Expression
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   ProcedureExpression                       -->
<!-- ____________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.ProcedureExpression
(Foundation.Data_Types.Expression.language,

Foundation.Data_Types.Expression.body,
                                                    XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.ProcedureExpression
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   ObjectSetExpression                       -->
<!-- ____________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.ObjectSetExpression
(Foundation.Data_Types.Expression.language,

Foundation.Data_Types.Expression.body,
                                                    XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.ObjectSetExpression
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   TimeExpression                            -->
<!-- ____________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.TimeExpression
(Foundation.Data_Types.Expression.language,

Foundation.Data_Types.Expression.body,
```

```
                                                                XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.TimeExpression
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   BooleanExpression                                   -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Foundation.Data_Types.BooleanExpression
(Foundation.Data_Types.Expression.language,

Foundation.Data_Types.Expression.body,
                                                          XMI.extension*)? >
<!ATTLIST Foundation.Data_Types.BooleanExpression
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Foundation.Data_Types ((Foundation.Data_Types.Enumeration |
                                    Foundation.Data_Types.EnumerationLiteral |
                                      Foundation.Data_Types.Primitive |
                                      Foundation.Data_Types.Structure |
                                     Foundation.Data_Types.MultiplicityRange |
                                      Foundation.Data_Types.Geometry |
                                      Foundation.Data_Types.GraphicMarker |
                                      Foundation.Data_Types.Mapping |
                                      Foundation.Data_Types.Expression |
                                   Foundation.Data_Types.ProcedureExpression |
                                   Foundation.Data_Types.ObjectSetExpression |
                                     Foundation.Data_Types.TimeExpression |
                                    Foundation.Data_Types.BooleanExpression)*)
                                       >
<!ATTLIST Foundation.Data_Types
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Foundation ((Foundation.Auxiliary_Elements | Foundation.Core |
                           Foundation.Extension_Mechanisms |
                           Foundation.Data_Types)*) >
<!ATTLIST Foundation
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL PACKAGE:   Model_Management                                  -->
<!-- ____________________________________________________________________ -->
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Package                                           -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Model_Management.Package.elementReference

(Model_Management.ElementReference)* >

<!ELEMENT Model_Management.Package (Foundation.Core.ModelElement.name,
                                    Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                       XMI.extension*,
                                  Foundation.Core.ModelElement.constraint*,
                                 Foundation.Core.ModelElement.requirement*,
                                   Foundation.Core.ModelElement.provision*,
                                  Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                    Foundation.Core.ModelElement.partition?,
                                     Foundation.Core.ModelElement.template?,
                                       Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                    Foundation.Core.ModelElement.namespace?,
                                     Foundation.Core.ModelElement.behavior*,
                                      Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Model_Management.Package.elementReference*,
                                  Foundation.Core.ModelElement.taggedValue*,
                                   Foundation.Core.Namespace.ownedElement*)?
                                        >
<!ATTLIST Model_Management.Package
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Subsystem                                         -->
<!-- ____________________________________________________________________ -->
```

```
<!ELEMENT Model_Management.Subsystem.isInstantiable EMPTY >
<!ATTLIST Model_Management.Subsystem.isInstantiable
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Model_Management.Subsystem (Foundation.Core.ModelElement.name,
                                      Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,

Model_Management.Subsystem.isInstantiable,
                                           XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,
                                      Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
                                      Foundation.Core.ModelElement.partition?,
                                       Foundation.Core.ModelElement.template?,
                                           Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
                                      Foundation.Core.ModelElement.namespace?,
                                       Foundation.Core.ModelElement.behavior*,
                                        Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
                                         Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,
                                      Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,
                                          Foundation.Core.Classifier.instance*,
```

```
Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,
                                        Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Model_Management.Package.elementReference*,

Foundation.Core.ModelElement.taggedValue*,
                                        Foundation.Core.Namespace.ownedElement*,
                                           Foundation.Core.Classifier.feature*)?
                                               >
<!ATTLIST Model_Management.Subsystem
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Model                                             -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Model_Management.Model (Foundation.Core.ModelElement.name,
                                        Foundation.Core.ModelElement.visibility,
                                    Foundation.Core.GeneralizableElement.isRoot,
                                    Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                         XMI.extension*,
                                      Foundation.Core.ModelElement.constraint*,
                                     Foundation.Core.ModelElement.requirement*,
                                      Foundation.Core.ModelElement.provision*,
                                     Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,
                                    Foundation.Core.ModelElement.collaboration*,
                                        Foundation.Core.ModelElement.partition?,
                                         Foundation.Core.ModelElement.template?,
                                         Foundation.Core.ModelElement.view*,
                                   Foundation.Core.ModelElement.presentation*,
                                        Foundation.Core.ModelElement.namespace?,
                                         Foundation.Core.ModelElement.behavior*,
                                         Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,
                                     Model_Management.Package.elementReference*,
                                      Foundation.Core.ModelElement.taggedValue*,
                                       Foundation.Core.Namespace.ownedElement*)?
```

```
                                                >
<!ATTLIST Model_Management.Model
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   ElementReference                                  -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Model_Management.ElementReference.visibility EMPTY >
<!ATTLIST Model_Management.ElementReference.visibility
              xmi.value (public | protected | private) #REQUIRED
>

<!ELEMENT Model_Management.ElementReference.alias (#PCDATA |
                                                   XMI.reference)* >

<!ELEMENT Model_Management.ElementReference.referencedElement
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |
```

```
Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |
```

```
Behavioral_Elements.Common_Behavior.ActionSequence |
Behavioral_Elements.Common_Behavior.Action |
Behavioral_Elements.Common_Behavior.CreateAction |
Behavioral_Elements.Common_Behavior.CallAction |
Behavioral_Elements.Common_Behavior.LocalInvocation |
Behavioral_Elements.Common_Behavior.ReturnAction |
Behavioral_Elements.Common_Behavior.SendAction |
Behavioral_Elements.Common_Behavior.UninterpretedAction |
Behavioral_Elements.Common_Behavior.TerminateAction |
Behavioral_Elements.Common_Behavior.DestroyAction |
Behavioral_Elements.Common_Behavior.Link |
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
```

```
Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)*
                                                                        >

<!ELEMENT Model_Management.ElementReference.package (Model_Management.Package |

Model_Management.Subsystem |
                                                    Model_Management.Model)*
                                                            >


<!ELEMENT Model_Management.ElementReference
(Model_Management.ElementReference.visibility,

Model_Management.ElementReference.alias,
                                                  XMI.extension*,

Model_Management.ElementReference.referencedElement*,

Model_Management.ElementReference.package*)?
                                                   >
<!ATTLIST Model_Management.ElementReference
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Model_Management ((Model_Management.Package |
                                  Model_Management.Subsystem |
                                  Model_Management.Model |
                                  Model_Management.ElementReference)*) >
<!ATTLIST Model_Management
              %XMI.element.att;
              %XMI.link.att;
>
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL PACKAGE:   Behavioral_Elements                               -->
<!-- ____________________________________________________________________ -->


<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL PACKAGE:   Common_Behavior                                   -->
<!-- ____________________________________________________________________ -->


<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   Request                                             -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Request.action
(Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction)*
                                                                        >

<!ELEMENT Behavioral_Elements.Common_Behavior.Request.messageInstance

(Behavioral_Elements.Common_Behavior.MessageInstance)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Request
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,
```

```
Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Request.action*,

Behavioral_Elements.Common_Behavior.Request.messageInstance*,

Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.Request
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Signal                                            -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Signal.reception

(Behavioral_Elements.Common_Behavior.Reception)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Signal.occurrence

(Behavioral_Elements.State_Machines.SignalEvent)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Signal
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,
```

```
Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Behavioral_Elements.Common_Behavior.Request.action*,

Behavioral_Elements.Common_Behavior.Request.messageInstance*,

Behavioral_Elements.Common_Behavior.Signal.reception*,

Behavioral_Elements.Common_Behavior.Signal.occurrence*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Behavioral_Elements.Common_Behavior.Signal.parameter*)?
                                                                >
<!ATTLIST Behavioral_Elements.Common_Behavior.Signal
              %XMI.element.att;
              %XMI.link.att;
```

```
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Exception                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Exception.context
(Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception)*
                                                                        >

<!ELEMENT Behavioral_Elements.Common_Behavior.Exception
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,
```

```
Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Behavioral_Elements.Common_Behavior.Request.action*,

Behavioral_Elements.Common_Behavior.Request.messageInstance*,

Behavioral_Elements.Common_Behavior.Signal.reception*,

Behavioral_Elements.Common_Behavior.Signal.occurrence*,

Behavioral_Elements.Common_Behavior.Exception.context*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Behavioral_Elements.Common_Behavior.Signal.parameter*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.Exception
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Reception                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Reception.isPolymorphic
                                                                      EMPTY
>
<!ATTLIST Behavioral_Elements.Common_Behavior.Reception.isPolymorphic
              xmi.value (false | true) #REQUIRED
>

<!ELEMENT Behavioral_Elements.Common_Behavior.Reception.application (#PCDATA |

XMI.reference)*
                                                                        >

<!ELEMENT Behavioral_Elements.Common_Behavior.Reception.signal
(Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception)
                                                                      >

<!ELEMENT Behavioral_Elements.Common_Behavior.Reception
(Foundation.Core.ModelElement.name,
```

```
        Foundation.Core.ModelElement.visibility,

        Foundation.Core.Feature.ownerScope,

        Foundation.Core.BehavioralFeature.isQuery,

        Behavioral_Elements.Common_Behavior.Reception.isPolymorphic,

        Behavioral_Elements.Common_Behavior.Reception.application,
                                                                    XMI.extension*,

        Foundation.Core.ModelElement.constraint*,

        Foundation.Core.ModelElement.requirement*,

        Foundation.Core.ModelElement.provision*,

        Foundation.Core.ModelElement.stereotype*,

        Foundation.Core.ModelElement.elementReference*,

        Foundation.Core.ModelElement.collaboration*,

        Foundation.Core.ModelElement.partition?,

        Foundation.Core.ModelElement.template?,

        Foundation.Core.ModelElement.view*,

        Foundation.Core.ModelElement.presentation*,

        Foundation.Core.ModelElement.namespace?,

        Foundation.Core.ModelElement.behavior*,

        Foundation.Core.ModelElement.binding?,

        Foundation.Core.ModelElement.implementation*,

        Foundation.Core.Feature.owner?,

        Foundation.Core.Feature.classifierRole*,

        Foundation.Core.BehavioralFeature.raisedException*,

        Behavioral_Elements.Common_Behavior.Reception.signal,

        Foundation.Core.ModelElement.taggedValue*,

        Foundation.Core.BehavioralFeature.parameter*)?
                                                                          >
        <!ATTLIST Behavioral_Elements.Common_Behavior.Reception
                    %XMI.element.att;
                    %XMI.link.att;
```

```
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Argument                                          -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Argument.value

(Foundation.Data_Types.Expression) >

<!ELEMENT Behavioral_Elements.Common_Behavior.Argument.action
(Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction)
                                                                      >

<!ELEMENT Behavioral_Elements.Common_Behavior.Argument
(Behavioral_Elements.Common_Behavior.Argument.value,
                                                    XMI.extension*,

Behavioral_Elements.Common_Behavior.Argument.action?)?
                                                       >
<!ATTLIST Behavioral_Elements.Common_Behavior.Argument
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   ActionSequence                                    -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.ActionSequence.transition

(Behavioral_Elements.State_Machines.Transition) >

<!ELEMENT Behavioral_Elements.Common_Behavior.ActionSequence.state
(Behavioral_Elements.State_Machines.State |
```

```
Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)
                                                                    >

<!ELEMENT Behavioral_Elements.Common_Behavior.ActionSequence.state2
(Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)
                                                                    >

<!ELEMENT Behavioral_Elements.Common_Behavior.ActionSequence
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                  XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
```

```
Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.ActionSequence.transition?,

Behavioral_Elements.Common_Behavior.ActionSequence.state?,

Behavioral_Elements.Common_Behavior.ActionSequence.state2?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.ActionSequence.action*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.ActionSequence
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   Action                                      -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Action.request
(Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Foundation.Core.Operation)
                                                                    >

<!ELEMENT Behavioral_Elements.Common_Behavior.Action.message

(Behavioral_Elements.Collaborations.Message) >

<!ELEMENT Behavioral_Elements.Common_Behavior.Action.actionSequence

(Behavioral_Elements.Common_Behavior.ActionSequence) >

<!ELEMENT Behavioral_Elements.Common_Behavior.Action
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,
```

```
Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                >
<!ATTLIST Behavioral_Elements.Common_Behavior.Action
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   CreateAction                                -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.CreateAction.instantiation
(Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |
```

```
Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)
                                                                        >

<!ELEMENT Behavioral_Elements.Common_Behavior.CreateAction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,
```

```
Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Behavioral_Elements.Common_Behavior.CreateAction.instantiation,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.CreateAction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   CallAction                                        -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.CallAction.mode EMPTY >
<!ATTLIST Behavioral_Elements.Common_Behavior.CallAction.mode
              xmi.value (synchronous | asynchronous) #REQUIRED
>

<!ELEMENT Behavioral_Elements.Common_Behavior.CallAction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Behavioral_Elements.Common_Behavior.CallAction.mode,
                                                          XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,
```

```
Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.CallAction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   LocalInvocation                             -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.LocalInvocation
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
```

```
Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.LocalInvocation
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL CLASS:   ReturnAction                              -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.ReturnAction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
```

```
Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.ReturnAction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL CLASS:   SendAction                                -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.SendAction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
```

```
Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.SendAction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   UninterpretedAction                       -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.UninterpretedAction.body (#PCDATA
|

XMI.reference)*
                                                                          >

<!ELEMENT Behavioral_Elements.Common_Behavior.UninterpretedAction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Behavioral_Elements.Common_Behavior.UninterpretedAction.body,

XMI.extension*,

Foundation.Core.ModelElement.constraint*,
```

```
Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.UninterpretedAction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   TerminateAction                           -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.TerminateAction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

XMI.extension*,
```

```
Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.TerminateAction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   DestroyAction                               -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.DestroyAction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,
```

```
Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Action.request?,

Behavioral_Elements.Common_Behavior.Action.message?,

Behavioral_Elements.Common_Behavior.Action.actionSequence?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Action.actualArgument*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.DestroyAction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   Link                                        -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Link.association
(Foundation.Core.Association |

Foundation.Core.AssociationClass |
```

```
Behavioral_Elements.Collaborations.AssociationRole)
                                                                        >

<!ELEMENT Behavioral_Elements.Common_Behavior.Link
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Link.association,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Link.linkRole*)?
                                                        >
<!ATTLIST Behavioral_Elements.Common_Behavior.Link
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   LinkEnd                                           -->
<!-- ____________________________________________________________________ -->
```

```
<!ELEMENT Behavioral_Elements.Common_Behavior.LinkEnd.associationEnd
(Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole)
                                                                    >

<!ELEMENT Behavioral_Elements.Common_Behavior.LinkEnd.instance
(Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance)
                                                                  >

<!ELEMENT Behavioral_Elements.Common_Behavior.LinkEnd.link
(Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject)
                                                                >

<!ELEMENT Behavioral_Elements.Common_Behavior.LinkEnd
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,
```

```
Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.LinkEnd.associationEnd,

Behavioral_Elements.Common_Behavior.LinkEnd.instance,

Behavioral_Elements.Common_Behavior.LinkEnd.link?,

Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.LinkEnd
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Instance                                          -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Instance.linkEnd

(Behavioral_Elements.Common_Behavior.LinkEnd)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Instance.messageInstance3

(Behavioral_Elements.Common_Behavior.MessageInstance)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Instance.messageInstance2

(Behavioral_Elements.Common_Behavior.MessageInstance)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Instance.attributeLink

(Behavioral_Elements.Common_Behavior.AttributeLink)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Instance.classifier
(Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |
```

```
Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)+
                                                                        >

<!ELEMENT Behavioral_Elements.Common_Behavior.Instance.messageInstance

(Behavioral_Elements.Common_Behavior.MessageInstance)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.Instance
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Instance.linkEnd*,
```

```
Behavioral_Elements.Common_Behavior.Instance.messageInstance3*,

Behavioral_Elements.Common_Behavior.Instance.messageInstance2*,

Behavioral_Elements.Common_Behavior.Instance.attributeLink*,

Behavioral_Elements.Common_Behavior.Instance.classifier+,

Behavioral_Elements.Common_Behavior.Instance.messageInstance*,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Instance.slot*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.Instance
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   AttributeLink                               -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.AttributeLink.attribute

(Foundation.Core.Attribute)* >

<!ELEMENT Behavioral_Elements.Common_Behavior.AttributeLink.value
(Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance)*
                                                                    >

<!ELEMENT Behavioral_Elements.Common_Behavior.AttributeLink.instance
(Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance)
                                                                    >
```

```
<!ELEMENT Behavioral_Elements.Common_Behavior.AttributeLink
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.AttributeLink.attribute*,

Behavioral_Elements.Common_Behavior.AttributeLink.value*,

Behavioral_Elements.Common_Behavior.AttributeLink.instance?,

Foundation.Core.ModelElement.taggedValue*)?
                                                                >
<!ATTLIST Behavioral_Elements.Common_Behavior.AttributeLink
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Object                                            -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.Object
(Foundation.Core.ModelElement.name,
```

```
Foundation.Core.ModelElement.visibility,
                                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Instance.linkEnd*,

Behavioral_Elements.Common_Behavior.Instance.messageInstance3*,

Behavioral_Elements.Common_Behavior.Instance.messageInstance2*,

Behavioral_Elements.Common_Behavior.Instance.attributeLink*,

Behavioral_Elements.Common_Behavior.Instance.classifier+,

Behavioral_Elements.Common_Behavior.Instance.messageInstance*,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Instance.slot*)?
                                                                >
<!ATTLIST Behavioral_Elements.Common_Behavior.Object
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
```

```
<!-- METAMODEL CLASS:   DataValue                                                -->
<!-- _____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.DataValue
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                                  XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Instance.linkEnd*,

Behavioral_Elements.Common_Behavior.Instance.messageInstance3*,

Behavioral_Elements.Common_Behavior.Instance.messageInstance2*,

Behavioral_Elements.Common_Behavior.Instance.attributeLink*,

Behavioral_Elements.Common_Behavior.Instance.classifier+,

Behavioral_Elements.Common_Behavior.Instance.messageInstance*,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Instance.slot*)?
                                                                        >
<!ATTLIST Behavioral_Elements.Common_Behavior.DataValue
```

```
                %XMI.element.att;
                %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:  LinkObject                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.LinkObject
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.Instance.linkEnd*,

Behavioral_Elements.Common_Behavior.Instance.messageInstance3*,

Behavioral_Elements.Common_Behavior.Instance.messageInstance2*,

Behavioral_Elements.Common_Behavior.Instance.attributeLink*,

Behavioral_Elements.Common_Behavior.Instance.classifier+,

Behavioral_Elements.Common_Behavior.Instance.messageInstance*,
```

```
Behavioral_Elements.Common_Behavior.Link.association,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.Common_Behavior.Instance.slot*,

Behavioral_Elements.Common_Behavior.Link.linkRole*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.LinkObject
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:  MessageInstance                            -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Common_Behavior.MessageInstance.specification
(Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Foundation.Core.Operation)

>

<!ELEMENT Behavioral_Elements.Common_Behavior.MessageInstance.sender
(Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance)
                                                                    >

<!ELEMENT Behavioral_Elements.Common_Behavior.MessageInstance.receiver
(Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance)
                                                                    >
```

```
<!ELEMENT Behavioral_Elements.Common_Behavior.MessageInstance.argument
(Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance)*
                                                                    >

<!ELEMENT Behavioral_Elements.Common_Behavior.MessageInstance
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Common_Behavior.MessageInstance.specification,

Behavioral_Elements.Common_Behavior.MessageInstance.sender,

Behavioral_Elements.Common_Behavior.MessageInstance.receiver,

Behavioral_Elements.Common_Behavior.MessageInstance.argument*,
```

```
Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Common_Behavior.MessageInstance
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Behavioral_Elements.Common_Behavior
((Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Common_Behavior.Reception |

Behavioral_Elements.Common_Behavior.Argument |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkEnd |

Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.AttributeLink |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.MessageInstance)*)
                                                          >
```

```
<!ATTLIST Behavioral_Elements.Common_Behavior
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL PACKAGE:  Collaborations                                   -->
<!-- ____________________________________________________________________ -->


<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:  Collaboration                                      -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Collaborations.Collaboration.representedClassifier
(Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)

>

<!ELEMENT Behavioral_Elements.Collaborations.Collaboration.representedOperation

(Foundation.Core.Operation) >
```

```
<!ELEMENT Behavioral_Elements.Collaborations.Collaboration.constrainingElement
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |
```

```
Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |
```

```
Behavioral_Elements.Common_Behavior.Link |
Behavioral_Elements.Common_Behavior.LinkObject |
Behavioral_Elements.Common_Behavior.LinkEnd |
Behavioral_Elements.Common_Behavior.Instance |
Behavioral_Elements.Common_Behavior.Object |
Behavioral_Elements.Common_Behavior.DataValue |
Behavioral_Elements.Use_Cases.UseCaseInstance |
Behavioral_Elements.Common_Behavior.AttributeLink |
Behavioral_Elements.Common_Behavior.MessageInstance |
Behavioral_Elements.Collaborations.Interaction |
Behavioral_Elements.State_Machines.StateMachine |
Behavioral_Elements.State_Machines.ActivityModel |
Behavioral_Elements.State_Machines.Guard |
Behavioral_Elements.State_Machines.StateVertex |
Behavioral_Elements.State_Machines.PseudoState |
Behavioral_Elements.State_Machines.State |
Behavioral_Elements.State_Machines.CompositeState |
Behavioral_Elements.State_Machines.SimpleState |
Behavioral_Elements.State_Machines.ActivityState |
Behavioral_Elements.State_Machines.ActionState |
Behavioral_Elements.State_Machines.ObjectFlowState |
Behavioral_Elements.State_Machines.SubmachineState |
Behavioral_Elements.State_Machines.Transition |
Behavioral_Elements.State_Machines.Event |
Behavioral_Elements.State_Machines.SignalEvent |
Behavioral_Elements.State_Machines.CallEvent |
Behavioral_Elements.State_Machines.TimeEvent |
```

```
Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)*

>

<!ELEMENT Behavioral_Elements.Collaborations.Collaboration
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Collaborations.Collaboration.representedClassifier?,

Behavioral_Elements.Collaborations.Collaboration.representedOperation?,

Behavioral_Elements.Collaborations.Collaboration.constrainingElement*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Behavioral_Elements.Collaborations.Collaboration.interaction*)?
                                                                        >
<!ATTLIST Behavioral_Elements.Collaborations.Collaboration
              %XMI.element.att;
```

```
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Interaction                                       -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Collaborations.Interaction.context

(Behavioral_Elements.Collaborations.Collaboration) >

<!ELEMENT Behavioral_Elements.Collaborations.Interaction.message

(Behavioral_Elements.Collaborations.Message)+ >

<!ELEMENT Behavioral_Elements.Collaborations.Interaction
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.Collaborations.Interaction.context?,

Behavioral_Elements.Collaborations.Interaction.message+,
```

```
Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Collaborations.Interaction
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   AssociationRole                           -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Collaborations.AssociationRole.multiplicity
(#PCDATA |

XMI.reference)*

>

<!ELEMENT Behavioral_Elements.Collaborations.AssociationRole.base
(Foundation.Core.Association |

Foundation.Core.AssociationClass |

Behavioral_Elements.Collaborations.AssociationRole)
                                                                    >

<!ELEMENT Behavioral_Elements.Collaborations.AssociationRole
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,

Behavioral_Elements.Collaborations.AssociationRole.multiplicity,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,
```

```
Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Association.link*,

Foundation.Core.Association.associationEnd*,

Behavioral_Elements.Collaborations.AssociationRole.base,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Association.connection*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Collaborations.AssociationRole
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   AssociationEndRole                          -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Collaborations.AssociationEndRole.base
(Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole)
                                                                      >


<!ELEMENT Behavioral_Elements.Collaborations.AssociationEndRole
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
```

```
Foundation.Core.AssociationEnd.isNavigable,

Foundation.Core.AssociationEnd.isOrdered,

Foundation.Core.AssociationEnd.aggregation,

Foundation.Core.AssociationEnd.multiplicity,

Foundation.Core.AssociationEnd.changeable,

Foundation.Core.AssociationEnd.targetScope,

XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.AssociationEnd.type,

Foundation.Core.AssociationEnd.specification*,

Foundation.Core.AssociationEnd.linkEnd*,

Foundation.Core.AssociationEnd.associationEndRole*,

Foundation.Core.AssociationEnd.association?,

Behavioral_Elements.Collaborations.AssociationEndRole.base,
```

```
Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.AssociationEnd.qualifier*)?
                                                                        >
<!ATTLIST Behavioral_Elements.Collaborations.AssociationEndRole
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Message                                           -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Collaborations.Message.predecessor

(Behavioral_Elements.Collaborations.Message)* >

<!ELEMENT Behavioral_Elements.Collaborations.Message.message

(Behavioral_Elements.Collaborations.Message)* >

<!ELEMENT Behavioral_Elements.Collaborations.Message.action
(Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction)
                                                                    >

<!ELEMENT Behavioral_Elements.Collaborations.Message.receiver

(Behavioral_Elements.Collaborations.ClassifierRole) >

<!ELEMENT Behavioral_Elements.Collaborations.Message.sender

(Behavioral_Elements.Collaborations.ClassifierRole) >

<!ELEMENT Behavioral_Elements.Collaborations.Message.interaction

(Behavioral_Elements.Collaborations.Interaction)* >
```

```
<!ELEMENT Behavioral_Elements.Collaborations.Message (XMI.extension*,

Behavioral_Elements.Collaborations.Message.predecessor*,

Behavioral_Elements.Collaborations.Message.message*,

Behavioral_Elements.Collaborations.Message.action,

Behavioral_Elements.Collaborations.Message.receiver,

Behavioral_Elements.Collaborations.Message.sender,

Behavioral_Elements.Collaborations.Message.interaction*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Collaborations.Message
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:  ClassifierRole                               -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Collaborations.ClassifierRole.multiplicity
(#PCDATA |

XMI.reference)*
                                                                        >

<!ELEMENT Behavioral_Elements.Collaborations.ClassifierRole.message2

(Behavioral_Elements.Collaborations.Message)* >

<!ELEMENT Behavioral_Elements.Collaborations.ClassifierRole.message

(Behavioral_Elements.Collaborations.Message)* >

<!ELEMENT Behavioral_Elements.Collaborations.ClassifierRole.base
(Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |
```

```
Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)
                                                                                >

<!ELEMENT Behavioral_Elements.Collaborations.ClassifierRole.availableFeature
(Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception)*

>

<!ELEMENT Behavioral_Elements.Collaborations.ClassifierRole
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,

Behavioral_Elements.Collaborations.ClassifierRole.multiplicity,
                                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,
```

```
Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Behavioral_Elements.Collaborations.ClassifierRole.message2*,

Behavioral_Elements.Collaborations.ClassifierRole.message*,

Behavioral_Elements.Collaborations.ClassifierRole.base,
```

```
Behavioral_Elements.Collaborations.ClassifierRole.availableFeature*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Collaborations.ClassifierRole
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Behavioral_Elements.Collaborations
((Behavioral_Elements.Collaborations.Collaboration |

Behavioral_Elements.Collaborations.Interaction |

Behavioral_Elements.Collaborations.AssociationRole |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Collaborations.Message |

Behavioral_Elements.Collaborations.ClassifierRole)*)
                                                        >
<!ATTLIST Behavioral_Elements.Collaborations
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL PACKAGE:   Use_Cases                               -->
<!-- ____________________________________________________________ -->


<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL CLASS:   Actor                                     -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Use_Cases.Actor
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,
                                                    XMI.extension*,
```

```
Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,
```

```
Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*)?
                                                        >
<!ATTLIST Behavioral_Elements.Use_Cases.Actor
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   UseCase                                           -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Use_Cases.UseCase.extensionPoint
                                                                  (XMI.any) >

<!ELEMENT Behavioral_Elements.Use_Cases.UseCase
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,

Behavioral_Elements.Use_Cases.UseCase.extensionPoint,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,
```

```
Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*)?
                                                              >
<!ATTLIST Behavioral_Elements.Use_Cases.UseCase
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:  UseCaseInstance                                    -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.Use_Cases.UseCaseInstance
(Foundation.Core.ModelElement.name,
```

```
Foundation.Core.ModelElement.visibility,
                                                                    XMI.extension*,
Foundation.Core.ModelElement.constraint*,
Foundation.Core.ModelElement.requirement*,
Foundation.Core.ModelElement.provision*,
Foundation.Core.ModelElement.stereotype*,
Foundation.Core.ModelElement.elementReference*,
Foundation.Core.ModelElement.collaboration*,
Foundation.Core.ModelElement.partition?,
Foundation.Core.ModelElement.template?,
Foundation.Core.ModelElement.view*,
Foundation.Core.ModelElement.presentation*,
Foundation.Core.ModelElement.namespace?,
Foundation.Core.ModelElement.behavior*,
Foundation.Core.ModelElement.binding?,
Foundation.Core.ModelElement.implementation*,
Behavioral_Elements.Common_Behavior.Instance.linkEnd*,
Behavioral_Elements.Common_Behavior.Instance.messageInstance3*,
Behavioral_Elements.Common_Behavior.Instance.messageInstance2*,
Behavioral_Elements.Common_Behavior.Instance.attributeLink*,
Behavioral_Elements.Common_Behavior.Instance.classifier+,
Behavioral_Elements.Common_Behavior.Instance.messageInstance*,
Foundation.Core.ModelElement.taggedValue*,
Behavioral_Elements.Common_Behavior.Instance.slot*)?
                                                                    >
<!ATTLIST Behavioral_Elements.Use_Cases.UseCaseInstance
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Behavioral_Elements.Use_Cases ((Behavioral_Elements.Use_Cases.Actor |
```

```
                                                Behavioral_Elements.Use_Cases.UseCase
|

Behavioral_Elements.Use_Cases.UseCaseInstance)*)
                                                        >
<!ATTLIST Behavioral_Elements.Use_Cases
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL PACKAGE:   State_Machines                                    -->
<!-- ____________________________________________________________________ -->


<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   StateMachine                                        -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.StateMachine.submachineState

(Behavioral_Elements.State_Machines.SubmachineState)* >

<!ELEMENT Behavioral_Elements.State_Machines.StateMachine.context
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |
```

```
Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |
```

```
Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |

Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.LinkEnd |

Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance |

Behavioral_Elements.Common_Behavior.AttributeLink |

Behavioral_Elements.Common_Behavior.MessageInstance |

Behavioral_Elements.Collaborations.Interaction |

Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel |

Behavioral_Elements.State_Machines.Guard |

Behavioral_Elements.State_Machines.StateVertex |
```

```
Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)
                                                                        >

<!ELEMENT Behavioral_Elements.State_Machines.StateMachine
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,
```

```
Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateMachine.submachineState*,

Behavioral_Elements.State_Machines.StateMachine.context?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.StateMachine.top,

Behavioral_Elements.State_Machines.StateMachine.transitions*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.StateMachine
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   Guard                                       -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.Guard.transition

(Behavioral_Elements.State_Machines.Transition) >

<!ELEMENT Behavioral_Elements.State_Machines.Guard
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,
```

```
Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.Guard.transition?,

Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.Guard
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   StateVertex                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.StateVertex.outgoing

(Behavioral_Elements.State_Machines.Transition)* >

<!ELEMENT Behavioral_Elements.State_Machines.StateVertex.incoming

(Behavioral_Elements.State_Machines.Transition)* >

<!ELEMENT Behavioral_Elements.State_Machines.StateVertex.parent

(Behavioral_Elements.State_Machines.CompositeState) >

<!ELEMENT Behavioral_Elements.State_Machines.StateVertex
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,
```

```
Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,

Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.StateVertex
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   Transition                                        -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.Transition.source
(Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |
```

```
Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)
                                                                    >

<!ELEMENT Behavioral_Elements.State_Machines.Transition.target
(Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)
                                                                    >

<!ELEMENT Behavioral_Elements.State_Machines.Transition.statemachine
(Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel)
                                                                     >

<!ELEMENT Behavioral_Elements.State_Machines.Transition.state
(Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)
                                                                    >

<!ELEMENT Behavioral_Elements.State_Machines.Transition.trigger
(Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |
```

```
Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent)
                                                                        >

<!ELEMENT Behavioral_Elements.State_Machines.Transition
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.Transition.source,

Behavioral_Elements.State_Machines.Transition.target,

Behavioral_Elements.State_Machines.Transition.statemachine?,

Behavioral_Elements.State_Machines.Transition.state?,

Behavioral_Elements.State_Machines.Transition.trigger?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.Transition.guard?,

Behavioral_Elements.State_Machines.Transition.effect?)?
```

```
                                                                        >
<!ATTLIST Behavioral_Elements.State_Machines.Transition
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   PseudoState                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.PseudoState.kind EMPTY >
<!ATTLIST Behavioral_Elements.State_Machines.PseudoState.kind
              xmi.value (initial | deepHistory | shallowHistory | join | fork
| branch | final) #REQUIRED
>

<!ELEMENT Behavioral_Elements.State_Machines.PseudoState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Behavioral_Elements.State_Machines.PseudoState.kind,
                                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,
```

```
Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,

Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.PseudoState
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   State                                             -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.State.deferredEvent
(Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent)*
                                                                        >

<!ELEMENT Behavioral_Elements.State_Machines.State.classifierInState

(Behavioral_Elements.State_Machines.ClassifierInState)* >

<!ELEMENT Behavioral_Elements.State_Machines.State.stateMachine
(Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel)
                                                                        >

<!ELEMENT Behavioral_Elements.State_Machines.State
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,
```

```
Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,

Behavioral_Elements.State_Machines.State.deferredEvent*,

Behavioral_Elements.State_Machines.State.classifierInState*,

Behavioral_Elements.State_Machines.State.stateMachine?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.State.internalTransition*,

Behavioral_Elements.State_Machines.State.entry?,

Behavioral_Elements.State_Machines.State.exit?)?
                                                        >
<!ATTLIST Behavioral_Elements.State_Machines.State
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   CompositeState                                      -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.CompositeState.isConcurrent

EMPTY >
<!ATTLIST Behavioral_Elements.State_Machines.CompositeState.isConcurrent
              xmi.value (false | true) #REQUIRED
```

```
>

<!ELEMENT Behavioral_Elements.State_Machines.CompositeState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Behavioral_Elements.State_Machines.CompositeState.isConcurrent,
                                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,

Behavioral_Elements.State_Machines.State.deferredEvent*,

Behavioral_Elements.State_Machines.State.classifierInState*,

Behavioral_Elements.State_Machines.State.stateMachine?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.State.internalTransition*,

Behavioral_Elements.State_Machines.State.entry?,
```

```
Behavioral_Elements.State_Machines.State.exit?,

Behavioral_Elements.State_Machines.CompositeState.substate+)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.CompositeState
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   SimpleState                                 -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.SimpleState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                      XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,
```

```
Behavioral_Elements.State_Machines.State.deferredEvent*,

Behavioral_Elements.State_Machines.State.classifierInState*,

Behavioral_Elements.State_Machines.State.stateMachine?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.State.internalTransition*,

Behavioral_Elements.State_Machines.State.entry?,

Behavioral_Elements.State_Machines.State.exit?)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.SimpleState
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   SubmachineState                           -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.SubmachineState.stateMachine
(Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel)

>

<!ELEMENT Behavioral_Elements.State_Machines.SubmachineState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,
```

```
Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,

Behavioral_Elements.State_Machines.State.deferredEvent*,

Behavioral_Elements.State_Machines.State.classifierInState*,

Behavioral_Elements.State_Machines.State.stateMachine?,

Behavioral_Elements.State_Machines.SubmachineState.stateMachine,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.State.internalTransition*,

Behavioral_Elements.State_Machines.State.entry?,

Behavioral_Elements.State_Machines.State.exit?)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.SubmachineState
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   Event                                       -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.Event.state
(Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |
```

```
Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)*
                                                                    >

<!ELEMENT Behavioral_Elements.State_Machines.Event.transition

(Behavioral_Elements.State_Machines.Transition)* >

<!ELEMENT Behavioral_Elements.State_Machines.Event
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                    XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.Event.state*,

Behavioral_Elements.State_Machines.Event.transition*,

Foundation.Core.ModelElement.taggedValue*)?
                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.Event
              %XMI.element.att;
              %XMI.link.att;
```

```
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   SignalEvent                                       -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.SignalEvent.signal
(Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception)
                                                                        >

<!ELEMENT Behavioral_Elements.State_Machines.SignalEvent
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.Event.state*,

Behavioral_Elements.State_Machines.Event.transition*,

Behavioral_Elements.State_Machines.SignalEvent.signal,

Foundation.Core.ModelElement.taggedValue*)?
```

```
                                                                        >
<!ATTLIST Behavioral_Elements.State_Machines.SignalEvent
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   CallEvent                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.CallEvent.operation

(Foundation.Core.Operation) >

<!ELEMENT Behavioral_Elements.State_Machines.CallEvent
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                          XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.Event.state*,

Behavioral_Elements.State_Machines.Event.transition*,

Behavioral_Elements.State_Machines.CallEvent.operation,
```

```
Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.CallEvent
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   TimeEvent                                         -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.TimeEvent.duration

(Foundation.Data_Types.TimeExpression) >

<!ELEMENT Behavioral_Elements.State_Machines.TimeEvent
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Behavioral_Elements.State_Machines.TimeEvent.duration,
                                                          XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.Event.state*,
```

```
Behavioral_Elements.State_Machines.Event.transition*,

Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.TimeEvent
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS:   ChangeEvent                                       -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.ChangeEvent.changeExpression

(Foundation.Data_Types.BooleanExpression) >

<!ELEMENT Behavioral_Elements.State_Machines.ChangeEvent
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Behavioral_Elements.State_Machines.ChangeEvent.changeExpression,
                                                                XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,
```

```
Behavioral_Elements.State_Machines.Event.state*,

Behavioral_Elements.State_Machines.Event.transition*,

Foundation.Core.ModelElement.taggedValue*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.ChangeEvent
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   ActivityModel                                       -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.ActivityModel
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                          XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateMachine.submachineState*,

Behavioral_Elements.State_Machines.StateMachine.context?,
```

```
Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.StateMachine.top,

Behavioral_Elements.State_Machines.StateMachine.transitions*,

Behavioral_Elements.State_Machines.ActivityModel.partition*)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.ActivityModel
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL CLASS:  Partition                                  -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.Partition.contents
(Foundation.Core.ModelElement |

Foundation.Core.Namespace |

Foundation.Core.GeneralizableElement |

Foundation.Core.Classifier |

Foundation.Core.Interface |

Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |
```

```
Foundation.Auxiliary_Elements.Component |

Foundation.Core.Association |

Behavioral_Elements.Collaborations.AssociationRole |

Foundation.Extension_Mechanisms.Stereotype |

Model_Management.Package |

Model_Management.Model |

Behavioral_Elements.Common_Behavior.Signal |

Behavioral_Elements.Common_Behavior.Exception |

Behavioral_Elements.Collaborations.Collaboration |

Foundation.Core.Feature |

Foundation.Core.StructuralFeature |

Foundation.Core.Attribute |

Foundation.Core.BehavioralFeature |

Foundation.Core.Operation |

Foundation.Core.Method |

Behavioral_Elements.Common_Behavior.Reception |

Foundation.Core.Parameter |

Foundation.Core.Constraint |

Foundation.Core.Dependency |

Foundation.Auxiliary_Elements.Refinement |

Foundation.Auxiliary_Elements.Usage |

Foundation.Auxiliary_Elements.Trace |

Foundation.Auxiliary_Elements.Binding |

Foundation.Core.Generalization |

Foundation.Core.AssociationEnd |

Behavioral_Elements.Collaborations.AssociationEndRole |

Behavioral_Elements.Common_Behavior.Request |
```

```
Behavioral_Elements.Common_Behavior.ActionSequence |

Behavioral_Elements.Common_Behavior.Action |

Behavioral_Elements.Common_Behavior.CreateAction |

Behavioral_Elements.Common_Behavior.CallAction |

Behavioral_Elements.Common_Behavior.LocalInvocation |

Behavioral_Elements.Common_Behavior.ReturnAction |

Behavioral_Elements.Common_Behavior.SendAction |

Behavioral_Elements.Common_Behavior.UninterpretedAction |

Behavioral_Elements.Common_Behavior.TerminateAction |

Behavioral_Elements.Common_Behavior.DestroyAction |

Behavioral_Elements.Common_Behavior.Link |

Behavioral_Elements.Common_Behavior.LinkObject |

Behavioral_Elements.Common_Behavior.LinkEnd |

Behavioral_Elements.Common_Behavior.Instance |

Behavioral_Elements.Common_Behavior.Object |

Behavioral_Elements.Common_Behavior.DataValue |

Behavioral_Elements.Use_Cases.UseCaseInstance |

Behavioral_Elements.Common_Behavior.AttributeLink |

Behavioral_Elements.Common_Behavior.MessageInstance |

Behavioral_Elements.Collaborations.Interaction |

Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.ActivityModel |

Behavioral_Elements.State_Machines.Guard |

Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |
```

```
Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Foundation.Auxiliary_Elements.Comment)*
                                                                        >

<!ELEMENT Behavioral_Elements.State_Machines.Partition.activityModel

(Behavioral_Elements.State_Machines.ActivityModel) >

<!ELEMENT Behavioral_Elements.State_Machines.Partition (XMI.extension*,

Behavioral_Elements.State_Machines.Partition.contents*,

Behavioral_Elements.State_Machines.Partition.activityModel?)?
                                                              >
<!ATTLIST Behavioral_Elements.State_Machines.Partition
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:  ActivityState                                -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.ActivityState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,
```

```
Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,

Behavioral_Elements.State_Machines.State.deferredEvent*,

Behavioral_Elements.State_Machines.State.classifierInState*,

Behavioral_Elements.State_Machines.State.stateMachine?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.State.internalTransition*,

Behavioral_Elements.State_Machines.State.entry?,

Behavioral_Elements.State_Machines.State.exit?)?
                                                                    >

<!ATTLIST Behavioral_Elements.State_Machines.ActivityState
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL CLASS:   ActionState                               -->
```

```
<!-- ____________________________________________________________ -->

<!ELEMENT Behavioral_Elements.State_Machines.ActionState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                        XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,

Behavioral_Elements.State_Machines.StateVertex.parent?,

Behavioral_Elements.State_Machines.State.deferredEvent*,

Behavioral_Elements.State_Machines.State.classifierInState*,

Behavioral_Elements.State_Machines.State.stateMachine?,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.State.internalTransition*,

Behavioral_Elements.State_Machines.State.entry?,
```

```
Behavioral_Elements.State_Machines.State.exit?)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.ActionState
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS:   ObjectFlowState                                     -->
<!-- ____________________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.ObjectFlowState.typeState

(Behavioral_Elements.State_Machines.ClassifierInState) >

<!ELEMENT Behavioral_Elements.State_Machines.ObjectFlowState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,
                                                          XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,

Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Behavioral_Elements.State_Machines.StateVertex.outgoing*,

Behavioral_Elements.State_Machines.StateVertex.incoming*,
```

```
Behavioral_Elements.State_Machines.StateVertex.parent?,

Behavioral_Elements.State_Machines.State.deferredEvent*,

Behavioral_Elements.State_Machines.State.classifierInState*,

Behavioral_Elements.State_Machines.State.stateMachine?,

Behavioral_Elements.State_Machines.ObjectFlowState.typeState,

Foundation.Core.ModelElement.taggedValue*,

Behavioral_Elements.State_Machines.State.internalTransition*,

Behavioral_Elements.State_Machines.State.entry?,

Behavioral_Elements.State_Machines.State.exit?)?
                                                                    >
<!ATTLIST Behavioral_Elements.State_Machines.ObjectFlowState
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________ -->
<!--                                                                -->
<!-- METAMODEL CLASS:   ClassifierInState                          -->
<!-- ____________________________________________________________ -->


<!ELEMENT Behavioral_Elements.State_Machines.ClassifierInState.objectFlowState

(Behavioral_Elements.State_Machines.ObjectFlowState)* >

<!ELEMENT Behavioral_Elements.State_Machines.ClassifierInState.inState
(Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.SubmachineState)
                                                                   >

<!ELEMENT Behavioral_Elements.State_Machines.ClassifierInState.type
(Foundation.Core.Classifier |

Foundation.Core.Interface |
```

```
Foundation.Core.Class |

Foundation.Core.AssociationClass |

Foundation.Core.DataType |

Foundation.Data_Types.Enumeration |

Foundation.Data_Types.Primitive |

Foundation.Data_Types.Structure |

Model_Management.Subsystem |

Behavioral_Elements.Collaborations.ClassifierRole |

Behavioral_Elements.Use_Cases.Actor |

Behavioral_Elements.Use_Cases.UseCase |

Behavioral_Elements.State_Machines.ClassifierInState |

Foundation.Auxiliary_Elements.Node |

Foundation.Auxiliary_Elements.Component)
                                                                    >

<!ELEMENT Behavioral_Elements.State_Machines.ClassifierInState
(Foundation.Core.ModelElement.name,

Foundation.Core.ModelElement.visibility,

Foundation.Core.GeneralizableElement.isRoot,

Foundation.Core.GeneralizableElement.isLeaf,

Foundation.Core.GeneralizableElement.isAbstract,

XMI.extension*,

Foundation.Core.ModelElement.constraint*,

Foundation.Core.ModelElement.requirement*,

Foundation.Core.ModelElement.provision*,

Foundation.Core.ModelElement.stereotype*,

Foundation.Core.ModelElement.elementReference*,

Foundation.Core.ModelElement.collaboration*,

Foundation.Core.ModelElement.partition?,
```

```
Foundation.Core.ModelElement.template?,

Foundation.Core.ModelElement.view*,

Foundation.Core.ModelElement.presentation*,

Foundation.Core.ModelElement.namespace?,

Foundation.Core.ModelElement.behavior*,

Foundation.Core.ModelElement.binding?,

Foundation.Core.ModelElement.implementation*,

Foundation.Core.GeneralizableElement.generalization*,

Foundation.Core.GeneralizableElement.specialization*,

Foundation.Core.Classifier.parameter*,

Foundation.Core.Classifier.structuralFeature*,

Foundation.Core.Classifier.specification*,

Foundation.Core.Classifier.associationEnd*,

Foundation.Core.Classifier.participant*,

Foundation.Core.Classifier.createAction*,

Foundation.Core.Classifier.instance*,

Foundation.Core.Classifier.collaboration*,

Foundation.Core.Classifier.classifierRole*,

Foundation.Core.Classifier.realization*,

Foundation.Core.Classifier.classifierInState*,

Behavioral_Elements.State_Machines.ClassifierInState.objectFlowState*,

Behavioral_Elements.State_Machines.ClassifierInState.inState,

Behavioral_Elements.State_Machines.ClassifierInState.type,

Foundation.Core.ModelElement.taggedValue*,

Foundation.Core.Namespace.ownedElement*,

Foundation.Core.Classifier.feature*)?
                                                                    >
<!ATTLIST  Behavioral_Elements.State_Machines.ClassifierInState
               %XMI.element.att;
```

```
              %XMI.link.att;
>

<!ELEMENT Behavioral_Elements.State_Machines
((Behavioral_Elements.State_Machines.StateMachine |

Behavioral_Elements.State_Machines.Guard |

Behavioral_Elements.State_Machines.StateVertex |

Behavioral_Elements.State_Machines.Transition |

Behavioral_Elements.State_Machines.PseudoState |

Behavioral_Elements.State_Machines.State |

Behavioral_Elements.State_Machines.CompositeState |

Behavioral_Elements.State_Machines.SimpleState |

Behavioral_Elements.State_Machines.SubmachineState |

Behavioral_Elements.State_Machines.Event |

Behavioral_Elements.State_Machines.SignalEvent |

Behavioral_Elements.State_Machines.CallEvent |

Behavioral_Elements.State_Machines.TimeEvent |

Behavioral_Elements.State_Machines.ChangeEvent |

Behavioral_Elements.State_Machines.ActivityModel |

Behavioral_Elements.State_Machines.Partition |

Behavioral_Elements.State_Machines.ActivityState |

Behavioral_Elements.State_Machines.ActionState |

Behavioral_Elements.State_Machines.ObjectFlowState |

Behavioral_Elements.State_Machines.ClassifierInState)*)
                                                        >
<!ATTLIST Behavioral_Elements.State_Machines
              %XMI.element.att;
              %XMI.link.att;
>

<!ELEMENT Behavioral_Elements ((Behavioral_Elements.Common_Behavior |
                                Behavioral_Elements.Collaborations |
                                Behavioral_Elements.Use_Cases |
                                Behavioral_Elements.State_Machines)*) >
<!ATTLIST Behavioral_Elements
              %XMI.element.att;
```

```
                    %XMI.link.att;
        >
```

MOF 1.1 DTD B

B.1 Introduction

Appendix B contains a DTD generated by programmatically applying Rule Set 3 of the DTD Production Rules to a MOF model of itself. This model represents the MOF model which is described in Section 3 MOF Model and Interfaces of the MOF 1.1 specification . The MOF model is used for describing OMG compliant metamodels, but for the purposes of transfering metamodels, the MOF model itself is treated as a metamodeland therefore has a corresponding DTD. This DTD generally follows the specification of the above section on representing metamodel information. By examining this DTD, you can gain a better understanding of the types of metamodel information that can be represented in an XML DTD, and the information that cannot be specified.

The structure of the DTD closely corresponds to the document "MOF 1.1 Specification, 1 September 1997". Each XML element corresponding to a MOF class has a comment indicating which section and pages of that document describe the class. You can verify the accuracy of the DTD against the document by reading the pages of the document in the comments and verifying that the encoding for them is correct.

B.2 MOF DTD

```
<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- XMI is the top-level XML element for XMI transfer text       -->
<!-- ____________________________________________________________ -->

<!ELEMENT XMI (XMI.header, XMI.content?, XMI.difference*,
                  XMI.extensions*) >
<!ATTLIST XMI
              xmi.version CDATA #FIXED "1.0"
              timestamp CDATA #IMPLIED
              verified (true | false) #IMPLIED
>
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- XMI.header contains documentation and identifies the model,      -->
<!-- metamodel, and metametamodel                                           -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.header (XMI.documentation?, XMI.model*, XMI.metamodel*,
                      XMI.metametamodel*) >

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- documentation for transfer data                                        -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.documentation (#PCDATA | XMI.owner | XMI.contact |
                             XMI.longDescription | XMI.shortDescription |
                             XMI.exporter | XMI.exporterVersion |
                             XMI.notice)* >

<!ELEMENT XMI.owner ANY >

<!ELEMENT XMI.contact ANY >

<!ELEMENT XMI.longDescription ANY >

<!ELEMENT XMI.shortDescription ANY >

<!ELEMENT XMI.exporter ANY >

<!ELEMENT XMI.exporterVersion ANY >

<!ELEMENT XMI.exporterID ANY >

<!ELEMENT XMI.notice ANY >

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- XMI.element.att defines the attributes that each XML element     -->
<!-- that corresponds to a metamodel class must have to conform to    -->
<!-- the XMI specification.                                                 -->
<!-- ____________________________________________________________________ -->

<!ENTITY % XMI.element.att
            'xmi.id ID #IMPLIED xmi.label CDATA #IMPLIED xmi.uuid
             CDATA #IMPLIED ' >

<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- XMI.link.att defines the attributes that each XML element that   -->
<!-- corresponds to a metamodel class must have to enable it to       -->
<!-- function as a simple XLink as well as refer to model             -->
<!-- constructs within the same XMI file.                                   -->
<!-- ____________________________________________________________________ -->
```

```
<!ENTITY % XMI.link.att
                  'xml:link CDATA #IMPLIED inline (true | false) #IMPLIED
                   actuate (show | user) #IMPLIED href CDATA #IMPLIED role
                   CDATA #IMPLIED title CDATA #IMPLIED show (embed | replace
                   | new) #IMPLIED behavior CDATA #IMPLIED xmi.idref IDREF
                   #IMPLIED xmi.uuidref CDATA #IMPLIED' >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.model identifies the model(s) being transferred                  -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.model ANY >
<!ATTLIST XMI.model
              %XMI.link.att;
              xmi.name      CDATA #REQUIRED
              xmi.version   CDATA #IMPLIED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.metamodel identifies the metamodel(s) for the transferred        -->
<!-- data                                                                 -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.metamodel ANY >
<!ATTLIST XMI.metamodel
              %XMI.link.att;
              xmi.name      CDATA #REQUIRED
              xmi.version   CDATA #IMPLIED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.metametamodel identifies the metametamodel(s) for the            -->
<!-- transferred data                                                     -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.metametamodel ANY >
<!ATTLIST XMI.metametamodel
              %XMI.link.att;
              xmi.name      CDATA #REQUIRED
              xmi.version   CDATA #IMPLIED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.content is the actual data being transferred                     -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.content ANY >

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.extensions contains data to transfer that does not conform       -->
```

```
<!-- to the metamodel(s) in the header                                    -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.extensions ANY >
<!ATTLIST XMI.extensions
              xmi.extender CDATA #REQUIRED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- extension contains information related to a specific model       -->
<!-- construct that is not defined in the metamodel(s) in the header -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.extension ANY >
<!ATTLIST XMI.extension
              %XMI.element.att;
              %XMI.link.att;
              xmi.extender    CDATA #REQUIRED
              xmi.extenderID CDATA #REQUIRED
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.difference holds XML elements representing differences to a -->
<!-- base model                                                           -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.difference (XMI.difference | XMI.delete | XMI.add |
                              XMI.replace)* >
<!ATTLIST XMI.difference
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.delete represents a deletion from a base model                -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.delete EMPTY >
<!ATTLIST XMI.delete
              %XMI.element.att;
              %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.add represents an addition to a base model                     -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.add ANY >
<!ATTLIST XMI.add
              %XMI.element.att;
              %XMI.link.att;
```

```
                xmi.position CDATA "-1"
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.replace represents the replacement of a model construct      -->
<!-- with another model construct in a base model                        -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.replace ANY >
<!ATTLIST XMI.replace
                %XMI.element.att;
                %XMI.link.att;
                xmi.position CDATA "-1"
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- XMI.reference may be used to refer to data types not defined in -->
<!-- the metamodel                                                        -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.reference ANY >
<!ATTLIST XMI.reference
                %XMI.link.att;
>

<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- This section contains the declaration of XML elements             -->
<!-- representing data types                                                 -->
<!-- ____________________________________________________________________ -->

<!ELEMENT XMI.TypeDefinitions ANY >

<!ELEMENT XMI.field ANY >

<!ELEMENT XMI.seqItem ANY >

<!ELEMENT XMI.octetStream (#PCDATA) >

<!ELEMENT XMI.unionDiscrim ANY >

<!ELEMENT XMI.enum EMPTY >
<!ATTLIST XMI.enum
                xmi.value CDATA #REQUIRED
>

<!ELEMENT XMI.any ANY >
<!ATTLIST XMI.any
                %XMI.link.att;
                xmi.type CDATA #IMPLIED
                xmi.name CDATA #IMPLIED
>
```

```
<!ELEMENT XMI.CorbaTypeCode (XMI.CorbaTcAlias | XMI.CorbaTcStruct |
                             XMI.CorbaTcSequence | XMI.CorbaTcArray |
                             XMI.CorbaTcEnum | XMI.CorbaTcUnion |
                             XMI.CorbaTcExcept | XMI.CorbaTcString |
                             XMI.CorbaTcWstring | XMI.CorbaTcShort |
                             XMI.CorbaTcLong | XMI.CorbaTcUshort |
                             XMI.CorbaTcUlong | XMI.CorbaTcFloat |
                             XMI.CorbaTcDouble | XMI.CorbaTcBoolean |
                             XMI.CorbaTcChar | XMI.CorbaTcWchar |
                             XMI.CorbaTcOctet | XMI.CorbaTcAny |
                             XMI.CorbaTcTypeCode | XMI.CorbaTcPrincipal |
                             XMI.CorbaTcNull | XMI.CorbaTcVoid |
                             XMI.CorbaTcLongLong |
                             XMI.CorbaTcLongDouble) >
<!ATTLIST XMI.CorbaTypeCode
          %XMI.element.att;
>

<!ELEMENT XMI.CorbaTcAlias (XMI.CorbaTypeCode) >
<!ATTLIST XMI.CorbaTcAlias
          xmi.tcName CDATA #REQUIRED
          xmi.tcId   CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTcStruct (XMI.CorbaTcField)* >
<!ATTLIST XMI.CorbaTcStruct
          xmi.tcName CDATA #REQUIRED
          xmi.tcId   CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTcField (XMI.CorbaTypeCode) >
<!ATTLIST XMI.CorbaTcField
          xmi.tcName CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcSequence (XMI.CorbaTypeCode |
                               XMI.CorbaRecursiveType) >
<!ATTLIST XMI.CorbaTcSequence
          xmi.tcLength CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaRecursiveType EMPTY >
<!ATTLIST XMI.CorbaRecursiveType
          xmi.offset CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcArray (XMI.CorbaTypeCode) >
<!ATTLIST XMI.CorbaTcArray
          xmi.tcLength CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcObjRef EMPTY >
<!ATTLIST XMI.CorbaTcObjRef
          xmi.tcName CDATA #REQUIRED
          xmi.tcId   CDATA #IMPLIED
```

```
>

<!ELEMENT XMI.CorbaTcEnum (XMI.CorbaTcEnumLabel) >
<!ATTLIST XMI.CorbaTcEnum
              xmi.tcName CDATA #REQUIRED
              xmi.tcId   CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTcEnumLabel EMPTY >
<!ATTLIST XMI.CorbaTcEnumLabel
              xmi.tcName CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcUnionMbr (XMI.CorbaTypeCode, XMI.any) >
<!ATTLIST XMI.CorbaTcUnionMbr
              xmi.tcName CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcUnion (XMI.CorbaTypeCode, XMI.CorbaTcUnionMbr*) >
<!ATTLIST XMI.CorbaTcUnion
              xmi.tcName CDATA #REQUIRED
              xmi.tcId   CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTcExcept (XMI.CorbaTcField)* >
<!ATTLIST XMI.CorbaTcExcept
              xmi.tcName CDATA #REQUIRED
              xmi.tcId   CDATA #IMPLIED
>

<!ELEMENT XMI.CorbaTcString EMPTY >
<!ATTLIST XMI.CorbaTcString
              xmi.tcLength CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcWstring EMPTY >
<!ATTLIST XMI.CorbaTcWstring
              xmi.tcLength CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcFixed EMPTY >
<!ATTLIST XMI.CorbaTcFixed
              xmi.tcDigits CDATA #REQUIRED
              xmi.tcScale  CDATA #REQUIRED
>

<!ELEMENT XMI.CorbaTcShort EMPTY >

<!ELEMENT XMI.CorbaTcLong EMPTY >

<!ELEMENT XMI.CorbaTcUshort EMPTY >

<!ELEMENT XMI.CorbaTcUlong EMPTY >

<!ELEMENT XMI.CorbaTcFloat EMPTY >
```

```
<!ELEMENT XMI.CorbaTcDouble EMPTY >

<!ELEMENT XMI.CorbaTcBoolean EMPTY >

<!ELEMENT XMI.CorbaTcChar EMPTY >

<!ELEMENT XMI.CorbaTcWchar EMPTY >

<!ELEMENT XMI.CorbaTcOctet EMPTY >

<!ELEMENT XMI.CorbaTcAny EMPTY >

<!ELEMENT XMI.CorbaTcTypeCode EMPTY >

<!ELEMENT XMI.CorbaTcPrincipal EMPTY >

<!ELEMENT XMI.CorbaTcNull EMPTY >

<!ELEMENT XMI.CorbaTcVoid EMPTY >

<!ELEMENT XMI.CorbaTcLongLong EMPTY >

<!ELEMENT XMI.CorbaTcLongDouble EMPTY >


<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL PACKAGE: Mof                                       -->
<!--                                                              -->
<!-- ____________________________________________________________ -->

<!ENTITY % Mof.Model.VisibilityKind
             ' XMI.value ( public_vis|protected_vis|private_vis) #REQUIRED'>

<!ENTITY % Mof.Model.TristateKind
             ' XMI.value ( yes|no|dont_care) #REQUIRED'>

<!ENTITY % Mof.Model.ScopeKind
             ' XMI.value ( instance_level|classifier_level) #REQUIRED'>

<!ENTITY % Mof.Model.AggregationKind
             ' XMI.value ( none|shared|composite) #REQUIRED'>

<!ENTITY % Mof.Model.DirectionKind
             ' XMI.value ( in_dir|out_dir|inout_dir|return_dir) #REQUIRED'>


<!-- ____________________________________________________________ -->
<!--                                                              -->
<!-- METAMODEL PACKAGE: Mof.Model                                 -->
<!--                                                              -->
<!-- ____________________________________________________________ -->
```

```
<!-- *****     Mof.Model.Contains    *******   -->

<!ELEMENT Mof.Model.Namespace.contents ( Mof.Model.ModelElement
   |Mof.Model.Namespace
   |Mof.Model.Constraint
   |Mof.Model.GeneralizableElement
   |Mof.Model.Class
   |Mof.Model.TypedElement
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.TypeAlias
   |Mof.Model.Feature
   |Mof.Model.StructuralFeature
   |Mof.Model.MofAttribute
   |Mof.Model.Reference
   |Mof.Model.AssociationEnd
   |Mof.Model.BehavioralFeature
   |Mof.Model.Operation
   |Mof.Model.MofException
   |Mof.Model.Association
   |Mof.Model.Package
   |Mof.Model.Import
   |Mof.Model.Parameter
   |Mof.Model.Constant
   |Mof.Model.Tag) >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.ModelElement                              -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ENTITY % Mof.Model.ModelElement.VerifyResultKind
             ' XMI.value ( valid|published|invalid) #REQUIRED'>

<!ENTITY % Mof.Model.ModelElement.DepthKind
             ' XMI.value ( shallow|deep) #REQUIRED'>

<!ELEMENT Mof.Model.ModelElement.name (#PCDATA|XMI.reference)*>

<!ELEMENT Mof.Model.ModelElement.annotation (#PCDATA|XMI.reference)*>

<!ELEMENT Mof.Model.ModelElement.container (Mof.Model.Namespace
   |Mof.Model.GeneralizableElement
   |Mof.Model.Class
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.BehavioralFeature
   |Mof.Model.Operation
   |Mof.Model.MofException
   |Mof.Model.Association
```

```
   |Mof.Model.Package)?>

<!ELEMENT Mof.Model.ModelElement.constraints (Mof.Model.Constraint)*>

<!ENTITY % Mof.Model.ModelElementProperties '(Mof.Model.ModelElement.name
    ,Mof.Model.ModelElement.annotation  )' >

<!ENTITY % Mof.Model.ModelElementAssociations '(Mof.Model.ModelElement.container?
    ,Mof.Model.ModelElement.constraints*)' >

<!ELEMENT Mof.Model.ModelElement ( %Mof.Model.ModelElementProperties;
        ,(XMI.extension*,    %Mof.Model.ModelElementAssociations; ) )?>

<!ATTLIST Mof.Model.ModelElement %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Namespace                                 -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ENTITY % Mof.Model.NamespaceProperties '(%Mof.Model.ModelElementProperties;)' >

<!ENTITY % Mof.Model.NamespaceAssociations
'(%Mof.Model.ModelElementAssociations;)' >

<!ENTITY % Mof.Model.NamespaceCompositions '(Mof.Model.Namespace.contents*)' >

<!ELEMENT Mof.Model.Namespace ( %Mof.Model.NamespaceProperties;
        ,(XMI.extension*,    %Mof.Model.NamespaceAssociations; )
        ,   %Mof.Model.NamespaceCompositions; )?>

<!ATTLIST Mof.Model.Namespace %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Constraint                                -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ENTITY % Mof.Model.Constraint.EvaluationKind
             ' XMI.value ( immediate|deferred) #REQUIRED'>

<!ELEMENT Mof.Model.Constraint.expression (XMI.any)>

<!ELEMENT Mof.Model.Constraint.language (#PCDATA|XMI.reference)*>

<!ELEMENT Mof.Model.Constraint.evaluationPolicy EMPTY>
<!ATTLIST Mof.Model.Constraint.evaluationPolicy
%Mof.Model.Constraint.EvaluationKind;>
```

```
<!ELEMENT Mof.Model.Constraint.constrainedElts (Mof.Model.ModelElement
   |Mof.Model.Namespace
   |Mof.Model.Constraint
   |Mof.Model.GeneralizableElement
   |Mof.Model.Class
   |Mof.Model.TypedElement
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.TypeAlias
   |Mof.Model.Feature
   |Mof.Model.StructuralFeature
   |Mof.Model.MofAttribute
   |Mof.Model.Reference
   |Mof.Model.AssociationEnd
   |Mof.Model.BehavioralFeature
   |Mof.Model.Operation
   |Mof.Model.MofException
   |Mof.Model.Association
   |Mof.Model.Package
   |Mof.Model.Import
   |Mof.Model.Parameter
   |Mof.Model.Constant
   |Mof.Model.Tag)+>

<!ENTITY % Mof.Model.ConstraintProperties '(%Mof.Model.ModelElementProperties;
    ,Mof.Model.Constraint.expression
    ,Mof.Model.Constraint.language
    ,Mof.Model.Constraint.evaluationPolicy  )' >

<!ENTITY % Mof.Model.ConstraintAssociations
'(%Mof.Model.ModelElementAssociations;
    ,Mof.Model.Constraint.constrainedElts+)' >

<!ELEMENT Mof.Model.Constraint ( %Mof.Model.ConstraintProperties;
        ,(XMI.extension*,    %Mof.Model.ConstraintAssociations; ) )?>

<!ATTLIST Mof.Model.Constraint %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.GeneralizableElement                      -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.GeneralizableElement.isRoot EMPTY>
<!ATTLIST Mof.Model.GeneralizableElement.isRoot %Mof.Model.TristateKind;>

<!ELEMENT Mof.Model.GeneralizableElement.isLeaf EMPTY>
<!ATTLIST Mof.Model.GeneralizableElement.isLeaf %Mof.Model.TristateKind;>

<!ELEMENT Mof.Model.GeneralizableElement.isAbstract EMPTY>
<!ATTLIST Mof.Model.GeneralizableElement.isAbstract
          XMI.value ( true | false ) #REQUIRED>
```

```
<!ELEMENT Mof.Model.GeneralizableElement.visibility EMPTY>
<!ATTLIST Mof.Model.GeneralizableElement.visibility %Mof.Model.VisibilityKind;>

<!ELEMENT Mof.Model.GeneralizableElement.supertypes
(Mof.Model.GeneralizableElement
   |Mof.Model.Class
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.Association
   |Mof.Model.Package)*>

<!ENTITY % Mof.Model.GeneralizableElementProperties
'(%Mof.Model.NamespaceProperties;
    ,Mof.Model.GeneralizableElement.isRoot
    ,Mof.Model.GeneralizableElement.isLeaf
    ,Mof.Model.GeneralizableElement.isAbstract
    ,Mof.Model.GeneralizableElement.visibility  )' >

<!ENTITY % Mof.Model.GeneralizableElementAssociations
'(%Mof.Model.NamespaceAssociations;
    ,Mof.Model.GeneralizableElement.supertypes*)' >

<!ENTITY % Mof.Model.GeneralizableElementCompositions
'(%Mof.Model.NamespaceCompositions;)' >

<!ELEMENT Mof.Model.GeneralizableElement (
%Mof.Model.GeneralizableElementProperties;
        ,(XMI.extension*,    %Mof.Model.GeneralizableElementAssociations; )
        ,   %Mof.Model.GeneralizableElementCompositions; )?>

<!ATTLIST Mof.Model.GeneralizableElement %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Classifier                                -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ENTITY % Mof.Model.ClassifierProperties
'(%Mof.Model.GeneralizableElementProperties;)' >

<!ENTITY % Mof.Model.ClassifierAssociations
'(%Mof.Model.GeneralizableElementAssociations;)' >

<!ENTITY % Mof.Model.ClassifierCompositions
'(%Mof.Model.GeneralizableElementCompositions;)' >

<!ELEMENT Mof.Model.Classifier ( %Mof.Model.ClassifierProperties;
        ,(XMI.extension*,    %Mof.Model.ClassifierAssociations; )
        ,   %Mof.Model.ClassifierCompositions; )?>

<!ATTLIST Mof.Model.Classifier %XMI.element.att; %XMI.link.att; >
```

```
<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS: Mof.Model.Class                                       -->
<!--                                                                        -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Class.isSingleton EMPTY>
<!ATTLIST Mof.Model.Class.isSingleton
          XMI.value ( true | false ) #REQUIRED>

<!ENTITY % Mof.Model.ClassProperties '(%Mof.Model.ClassifierProperties;
   ,Mof.Model.Class.isSingleton  )' >

<!ENTITY % Mof.Model.ClassAssociations '(%Mof.Model.ClassifierAssociations;)' >

<!ENTITY % Mof.Model.ClassCompositions '(%Mof.Model.ClassifierCompositions;)' >

<!ELEMENT Mof.Model.Class ( %Mof.Model.ClassProperties;
        ,(XMI.extension*,    %Mof.Model.ClassAssociations; )
        ,   %Mof.Model.ClassCompositions; )?>

<!ATTLIST Mof.Model.Class %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS: Mof.Model.TypedElement                                -->
<!--                                                                        -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.TypedElement.type (Mof.Model.Class
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.Association) >

<!ENTITY % Mof.Model.TypedElementProperties
'(%Mof.Model.ModelElementProperties;)' >

<!ENTITY % Mof.Model.TypedElementAssociations
'(%Mof.Model.ModelElementAssociations;
   ,Mof.Model.TypedElement.type )' >

<!ELEMENT Mof.Model.TypedElement ( %Mof.Model.TypedElementProperties;
        ,(XMI.extension*,    %Mof.Model.TypedElementAssociations; ) )?>

<!ATTLIST Mof.Model.TypedElement %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                        -->
```

```
<!-- METAMODEL CLASS: Mof.Model.DataType                                    -->
<!--                                                                        -->
<!-- ______________________________________________________________________ -->

<!ELEMENT Mof.Model.DataType.typeCode (#PCDATA|XMI.reference)*>

<!ENTITY % Mof.Model.DataTypeProperties '(%Mof.Model.ClassifierProperties;
   ,Mof.Model.DataType.typeCode  )' >

<!ENTITY % Mof.Model.DataTypeAssociations '(%Mof.Model.ClassifierAssociations;)'
>

<!ENTITY % Mof.Model.DataTypeCompositions '(%Mof.Model.ClassifierCompositions;)'
>

<!ELEMENT Mof.Model.DataType ( %Mof.Model.DataTypeProperties;
        ,(XMI.extension*,    %Mof.Model.DataTypeAssociations; )
        ,   %Mof.Model.DataTypeCompositions; )?>

<!ATTLIST Mof.Model.DataType %XMI.element.att; %XMI.link.att; >



<!-- ______________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS: Mof.Model.TypeAlias                                   -->
<!--                                                                        -->
<!-- ______________________________________________________________________ -->

<!ELEMENT Mof.Model.TypeAlias.multiplicity (XMI.field|XMI.reference)*>

<!ENTITY % Mof.Model.TypeAliasProperties '(%Mof.Model.TypedElementProperties;
   ,Mof.Model.TypeAlias.multiplicity  )' >

<!ENTITY % Mof.Model.TypeAliasAssociations
'(%Mof.Model.TypedElementAssociations;)' >

<!ELEMENT Mof.Model.TypeAlias ( %Mof.Model.TypeAliasProperties;
        ,(XMI.extension*,    %Mof.Model.TypeAliasAssociations; ) )?>

<!ATTLIST Mof.Model.TypeAlias %XMI.element.att; %XMI.link.att; >



<!-- ______________________________________________________________________ -->
<!--                                                                        -->
<!-- METAMODEL CLASS: Mof.Model.Feature                                     -->
<!--                                                                        -->
<!-- ______________________________________________________________________ -->

<!ELEMENT Mof.Model.Feature.scope EMPTY>
<!ATTLIST Mof.Model.Feature.scope %Mof.Model.ScopeKind;>

<!ELEMENT Mof.Model.Feature.visibility EMPTY>
<!ATTLIST Mof.Model.Feature.visibility %Mof.Model.VisibilityKind;>
```

```
<!ENTITY % Mof.Model.FeatureProperties '(%Mof.Model.ModelElementProperties;
    ,Mof.Model.Feature.scope
    ,Mof.Model.Feature.visibility  )' >

<!ENTITY % Mof.Model.FeatureAssociations '(%Mof.Model.ModelElementAssociations;)'
>

<!ELEMENT Mof.Model.Feature ( %Mof.Model.FeatureProperties;
        ,(XMI.extension*,    %Mof.Model.FeatureAssociations; ) )?>

<!ATTLIST Mof.Model.Feature %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.StructuralFeature                         -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.StructuralFeature.multiplicity (XMI.field|XMI.reference)*>

<!ELEMENT Mof.Model.StructuralFeature.isChangeable EMPTY>
<!ATTLIST Mof.Model.StructuralFeature.isChangeable
         XMI.value ( true | false ) #REQUIRED>

<!ENTITY % Mof.Model.StructuralFeatureProperties
'(%Mof.Model.TypedElementProperties;
    ,Mof.Model.Feature.scope
    ,Mof.Model.Feature.visibility
    ,Mof.Model.StructuralFeature.multiplicity
    ,Mof.Model.StructuralFeature.isChangeable  )' >

<!ENTITY % Mof.Model.StructuralFeatureAssociations
'(%Mof.Model.TypedElementAssociations;)' >

<!ELEMENT Mof.Model.StructuralFeature ( %Mof.Model.StructuralFeatureProperties;
        ,(XMI.extension*,    %Mof.Model.StructuralFeatureAssociations; ) )?>

<!ATTLIST Mof.Model.StructuralFeature %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.MofAttribute                              -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.MofAttribute.isDerived EMPTY>
<!ATTLIST Mof.Model.MofAttribute.isDerived
         XMI.value ( true | false ) #REQUIRED>
```

```
<!ENTITY % Mof.Model.MofAttributeProperties
'(%Mof.Model.StructuralFeatureProperties;
   ,Mof.Model.MofAttribute.isDerived  )' >

<!ENTITY % Mof.Model.MofAttributeAssociations
'(%Mof.Model.StructuralFeatureAssociations;)' >

<!ELEMENT Mof.Model.MofAttribute ( %Mof.Model.MofAttributeProperties;
        ,(XMI.extension*,    %Mof.Model.MofAttributeAssociations; ) )?>

<!ATTLIST Mof.Model.MofAttribute %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Reference                                 -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Reference.referencedEnd (Mof.Model.AssociationEnd) >

<!ENTITY % Mof.Model.ReferenceProperties
'(%Mof.Model.StructuralFeatureProperties;)' >

<!ENTITY % Mof.Model.ReferenceAssociations
'(%Mof.Model.StructuralFeatureAssociations;
   ,Mof.Model.Reference.referencedEnd )' >

<!ELEMENT Mof.Model.Reference ( %Mof.Model.ReferenceProperties;
        ,(XMI.extension*,    %Mof.Model.ReferenceAssociations; ) )?>

<!ATTLIST Mof.Model.Reference %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.AssociationEnd                            -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.AssociationEnd.isNavigable EMPTY>
<!ATTLIST Mof.Model.AssociationEnd.isNavigable
         XMI.value ( true | false ) #REQUIRED>

<!ELEMENT Mof.Model.AssociationEnd.aggregation EMPTY>
<!ATTLIST Mof.Model.AssociationEnd.aggregation %Mof.Model.AggregationKind;>

<!ELEMENT Mof.Model.AssociationEnd.multiplicity (XMI.field|XMI.reference)*>

<!ELEMENT Mof.Model.AssociationEnd.isChangeable EMPTY>
<!ATTLIST Mof.Model.AssociationEnd.isChangeable
         XMI.value ( true | false ) #REQUIRED>
```

```
<!ENTITY % Mof.Model.AssociationEndProperties
'(%Mof.Model.TypedElementProperties;
   ,Mof.Model.AssociationEnd.isNavigable
   ,Mof.Model.AssociationEnd.aggregation
   ,Mof.Model.AssociationEnd.multiplicity
   ,Mof.Model.AssociationEnd.isChangeable  )' >

<!ENTITY % Mof.Model.AssociationEndAssociations
'(%Mof.Model.TypedElementAssociations;)' >

<!ELEMENT Mof.Model.AssociationEnd ( %Mof.Model.AssociationEndProperties;
        ,(XMI.extension*,    %Mof.Model.AssociationEndAssociations; ) )?>

<!ATTLIST Mof.Model.AssociationEnd %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.BehavioralFeature                         -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ENTITY % Mof.Model.BehavioralFeatureProperties
'(%Mof.Model.NamespaceProperties;
   ,Mof.Model.Feature.scope
   ,Mof.Model.Feature.visibility  )' >

<!ENTITY % Mof.Model.BehavioralFeatureAssociations
'(%Mof.Model.NamespaceAssociations;)' >

<!ENTITY % Mof.Model.BehavioralFeatureCompositions
'(%Mof.Model.NamespaceCompositions;)' >

<!ELEMENT Mof.Model.BehavioralFeature ( %Mof.Model.BehavioralFeatureProperties;
        ,(XMI.extension*,    %Mof.Model.BehavioralFeatureAssociations; )
        ,   %Mof.Model.BehavioralFeatureCompositions; )?>

<!ATTLIST Mof.Model.BehavioralFeature %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Operation                                 -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Operation.isQuery EMPTY>
<!ATTLIST Mof.Model.Operation.isQuery
         XMI.value ( true | false ) #REQUIRED>

<!ELEMENT Mof.Model.Operation.exceptions (Mof.Model.MofException)*>
```

```
<!ENTITY % Mof.Model.OperationProperties
'(%Mof.Model.BehavioralFeatureProperties;
   ,Mof.Model.Operation.isQuery   )' >

<!ENTITY % Mof.Model.OperationAssociations
'(%Mof.Model.BehavioralFeatureAssociations;
   ,Mof.Model.Operation.exceptions*)' >

<!ENTITY % Mof.Model.OperationCompositions
'(%Mof.Model.BehavioralFeatureCompositions;)' >

<!ELEMENT Mof.Model.Operation ( %Mof.Model.OperationProperties;
        ,(XMI.extension*,    %Mof.Model.OperationAssociations; )
        ,   %Mof.Model.OperationCompositions; )?>

<!ATTLIST Mof.Model.Operation %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.MofException                          -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ENTITY % Mof.Model.MofExceptionProperties
'(%Mof.Model.BehavioralFeatureProperties;)' >

<!ENTITY % Mof.Model.MofExceptionAssociations
'(%Mof.Model.BehavioralFeatureAssociations;)' >

<!ENTITY % Mof.Model.MofExceptionCompositions
'(%Mof.Model.BehavioralFeatureCompositions;)' >

<!ELEMENT Mof.Model.MofException ( %Mof.Model.MofExceptionProperties;
        ,(XMI.extension*,    %Mof.Model.MofExceptionAssociations; )
        ,   %Mof.Model.MofExceptionCompositions; )?>

<!ATTLIST Mof.Model.MofException %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Association                           -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Association.isDerived EMPTY>
<!ATTLIST Mof.Model.Association.isDerived
         XMI.value ( true | false ) #REQUIRED>

<!ENTITY % Mof.Model.AssociationProperties '(%Mof.Model.ClassifierProperties;
   ,Mof.Model.Association.isDerived   )' >
```

```
<!ENTITY % Mof.Model.AssociationAssociations
'(%Mof.Model.ClassifierAssociations;)' >

<!ENTITY % Mof.Model.AssociationCompositions
'(%Mof.Model.ClassifierCompositions;)' >

<!ELEMENT Mof.Model.Association ( %Mof.Model.AssociationProperties;
        ,(XMI.extension*,    %Mof.Model.AssociationAssociations; )
        ,   %Mof.Model.AssociationCompositions; )?>

<!ATTLIST Mof.Model.Association %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Package                                   -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ENTITY % Mof.Model.PackageProperties
'(%Mof.Model.GeneralizableElementProperties;)' >

<!ENTITY % Mof.Model.PackageAssociations
'(%Mof.Model.GeneralizableElementAssociations;)' >

<!ENTITY % Mof.Model.PackageCompositions
'(%Mof.Model.GeneralizableElementCompositions;)' >

<!ELEMENT Mof.Model.Package ( %Mof.Model.PackageProperties;
        ,(XMI.extension*,    %Mof.Model.PackageAssociations; )
        ,   %Mof.Model.PackageCompositions; )?>

<!ATTLIST Mof.Model.Package %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Import                                    -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Import.visibility EMPTY>
<!ATTLIST Mof.Model.Import.visibility %Mof.Model.VisibilityKind;>

<!ELEMENT Mof.Model.Import.importedNamespace (Mof.Model.Namespace
   |Mof.Model.GeneralizableElement
   |Mof.Model.Class
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.BehavioralFeature
   |Mof.Model.Operation
   |Mof.Model.MofException
   |Mof.Model.Association
```

```
   |Mof.Model.Package) >

<!ENTITY % Mof.Model.ImportProperties '(%Mof.Model.ModelElementProperties;
    ,Mof.Model.Import.visibility  )' >

<!ENTITY % Mof.Model.ImportAssociations '(%Mof.Model.ModelElementAssociations;
    ,Mof.Model.Import.importedNamespace )' >

<!ELEMENT Mof.Model.Import ( %Mof.Model.ImportProperties;
        ,(XMI.extension*,    %Mof.Model.ImportAssociations; ) )?>

<!ATTLIST Mof.Model.Import %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Parameter                                 -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Parameter.direction EMPTY>
<!ATTLIST Mof.Model.Parameter.direction %Mof.Model.DirectionKind;>

<!ELEMENT Mof.Model.Parameter.multiplicity (XMI.field|XMI.reference)*>

<!ENTITY % Mof.Model.ParameterProperties '(%Mof.Model.TypedElementProperties;
    ,Mof.Model.Parameter.direction
    ,Mof.Model.Parameter.multiplicity  )' >

<!ENTITY % Mof.Model.ParameterAssociations
'(%Mof.Model.TypedElementAssociations;)' >

<!ELEMENT Mof.Model.Parameter ( %Mof.Model.ParameterProperties;
        ,(XMI.extension*,    %Mof.Model.ParameterAssociations; ) )?>

<!ATTLIST Mof.Model.Parameter %XMI.element.att; %XMI.link.att; >



<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Constant                                  -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Constant.constValue (XMI.any)>

<!ENTITY % Mof.Model.ConstantProperties '(%Mof.Model.TypedElementProperties;
    ,Mof.Model.Constant.constValue  )' >

<!ENTITY % Mof.Model.ConstantAssociations
'(%Mof.Model.TypedElementAssociations;)' >

<!ELEMENT Mof.Model.Constant ( %Mof.Model.ConstantProperties;
```

```
        ,(XMI.extension*,    %Mof.Model.ConstantAssociations; ) )?>

<!ATTLIST Mof.Model.Constant %XMI.element.att; %XMI.link.att; >


<!-- ____________________________________________________________________ -->
<!--                                                                      -->
<!-- METAMODEL CLASS: Mof.Model.Tag                                       -->
<!--                                                                      -->
<!-- ____________________________________________________________________ -->

<!ELEMENT Mof.Model.Tag.tagId (#PCDATA|XMI.reference)*>

<!ELEMENT Mof.Model.Tag.values (XMI.any)>

<!ELEMENT Mof.Model.Tag.elements (Mof.Model.ModelElement
   |Mof.Model.Namespace
   |Mof.Model.Constraint
   |Mof.Model.GeneralizableElement
   |Mof.Model.Class
   |Mof.Model.TypedElement
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.TypeAlias
   |Mof.Model.Feature
   |Mof.Model.StructuralFeature
   |Mof.Model.MofAttribute
   |Mof.Model.Reference
   |Mof.Model.AssociationEnd
   |Mof.Model.BehavioralFeature
   |Mof.Model.Operation
   |Mof.Model.MofException
   |Mof.Model.Association
   |Mof.Model.Package
   |Mof.Model.Import
   |Mof.Model.Parameter
   |Mof.Model.Constant
   |Mof.Model.Tag)+>

<!ENTITY % Mof.Model.TagProperties '(%Mof.Model.ModelElementProperties;
    ,Mof.Model.Tag.tagId
    ,Mof.Model.Tag.values  )' >

<!ENTITY % Mof.Model.TagAssociations '(%Mof.Model.ModelElementAssociations;
    ,Mof.Model.Tag.elements+)' >

<!ELEMENT Mof.Model.Tag ( %Mof.Model.TagProperties;
        ,(XMI.extension*,    %Mof.Model.TagAssociations; ) )?>

<!ATTLIST Mof.Model.Tag %XMI.element.att; %XMI.link.att; >

<!ELEMENT Mof.Model ((Mof.Model.ModelElement
   |Mof.Model.Namespace
   |Mof.Model.Constraint
```

```
   |Mof.Model.GeneralizableElement
   |Mof.Model.Class
   |Mof.Model.TypedElement
   |Mof.Model.Classifier
   |Mof.Model.DataType
   |Mof.Model.TypeAlias
   |Mof.Model.Feature
   |Mof.Model.StructuralFeature
   |Mof.Model.MofAttribute
   |Mof.Model.Reference
   |Mof.Model.AssociationEnd
   |Mof.Model.BehavioralFeature
   |Mof.Model.Operation
   |Mof.Model.MofException
   |Mof.Model.Association
   |Mof.Model.Package
   |Mof.Model.Import
   |Mof.Model.Parameter
   |Mof.Model.Constant
   |Mof.Model.Tag)*)>
<!ATTLIST Mof.Model %XMI.element.att; %XMI.link.att;>


<!ELEMENT Mof ((Mof.Model)*)>
<!ATTLIST Mof %XMI.element.att; %XMI.link.att;>
```

B.3 Example

This is an example MOF document which consists of a package containing a class that contains an attribute.

```
<?xml version="1.0" encoding=”ISO-8859-1”?>
<!DOCTYPE XMI SYSTEM "mof.dtd">

<XMI xmi.version="1.0">
   <XMI.header>
      <XMI.metamodel xmi.name="uml" xmi.version="1.1" />
   </XMI.header>
   <XMI.content>
      <MoF.Model.Package xmi.id="i00000001">
         <Mof.Model.ModelElement.name>package1</Mof.Model.ModelElement.name>
         <Mof.Model.ModelElement.annotation/>
         <Mof.Model.GeneralizableElement.isRoot XMI.value="yes"/>
         <Mof.Model.GeneralizableElement.isLeaf XMI.value="yes"/>
         <Mof.Model.GeneralizableElement.isAbstract XMI.value=”false”/>
         <Mof.Model.GeneralizableElement.visibility XMI.value="public_vis" />
         <Mof.Model.Namespace.contents>
            <Mof.Model.Class xmi.id="i00000002">
               <Mof.Model.ModelElement.name>class1</Mof.Model.ModelElement.name>
               <Mof.Model.ModelElement.annotation/>
               <Mof.Model.GeneralizableElement.isRoot XMI.value="yes"/>
```

```
                    <Mof.Model.GeneralizableElement.isLeaf XMI.value="yes"/>
                    <Mof.Model.GeneralizableElement.isAbstract XMI.value=”false”/>
                    <Mof.Model.GeneralizableElement.visibility
                          XMI.value="public_vis" />
                    <Mof.Model.Class.isSingleton XMI.value="false"/>
                    <Mof.Model.Namespace.contents>
                       <Mof.Model.MofAttribute xmi.id="i00000003">
                          <Mof.Model.ModelElement.name>
                             attribute1
                          </Mof.Model.ModelElement.name>
                          <Mof.Model.ModelElement.annotation/>
                          <Mof.Model.Feature.scope XMI.value = “instance_level” />
                          <Mof.Model.Feature.visibility XMI.value="public_vis" />
                          <Mof.Model.StructuralFeature.multiplicity>
                             <XMI.field>1</XMI.field>
                             <XMI.field>1</XMI.field>
                          </Mof.Model.StructuralFeature.multiplicity>
                          <Mof.Model.StructuralFeature.isChangeable
                                                      XMI.value=”true”/>
                          <Mof.Model.MofAttribute.isDerived XMI.value=”false”/>
                          <Mof.Model.TypedElement.type>
                             <Mof.Model.DataType xmi.idref = “i00000004”/>
                          </Mof.Model.TypedElement.type>
                       </Mof.Model.MofAttribute>
                    </Mof.Model.Namespace.contents>
                 </Mof.Model.Class>
              </Mof.Model.Namespace.contents>
              <Mof.Model.Namespace.contents>
                 <Mof.Model.DataType xmi.id="i00000004">
                    <Mof.Model.ModelElement.name>
                       integer
                    </Mof.Model.ModelElement.name>
                    <Mof.Model.ModelElement.annotation/>
                    <Mof.Model.GeneralizableElement.isRoot XMI.value="yes"/>
                    <Mof.Model.GeneralizableElement.isLeaf XMI.value="yes"/>
                    <Mof.Model.GeneralizableElement.isAbstract XMI.value="false"/>
                    <Mof.Model.GeneralizableElement.visibility
                          XMI.value="public_vis" />
                    <Mof.Model.DataType.typeCode>
                          tk_integer
                    </Mof.Model.DataType.typeCode>
                 </Mof.Model.DataType>
              </Mof.Model.Namespace.contents>
           </Mof.Model.Package>
        </XMI.content>
     </XMI>
```

Example Model Encodings C

C.1 Introduction

This appendix provides two extremely simple UML models and their resultant XML encoding.

C.2 Model1 Model

This is an example of a model containing a class which contains an attribute. It is encoded in XMI using the UML DTD specified in Appendix A..

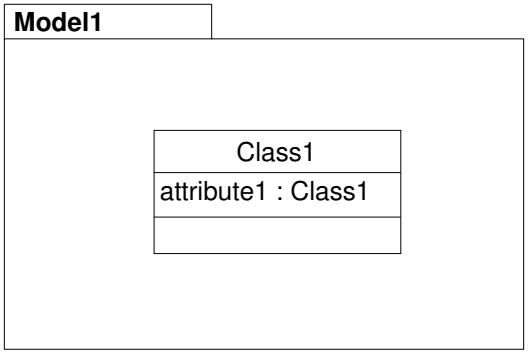

Figure C-1 Simple UML Model Model1

```
<?xml version = "1.0"?>
<!DOCTYPE XMI SYSTEM "uml.dtd">
<XMI xmi.version="1.0">
<XMI.header>
  <XMI.metamodel xmi.name="uml" xmi.version="1.1" />
  </XMI.header>
  <XMI.content>
  <Model_Management.Model xmi.id="i00000001">
    <Foundation.Core.ModelElement.name>Model1</Foundation.Core.ModelElement.name>
      <Foundation.Core.ModelElement.visibility xmi.value="public"/>
      <Foundation.Core.GeneralizableElement.isRoot xmi.value="true"/>
      <Foundation.Core.GeneralizableElement.isLeaf xmi.value="true"/>
      <Foundation.Core.GeneralizableElement.isAbstract xmi.value="false"/>
      <Foundation.Core.Namespace.ownedElement>
        <Foundation.Core.Class xmi.id="i00000002">
          <Foundation.Core.ModelElement.name>
             class1
          </Foundation.Core.ModelElement.name>
          <Foundation.Core.ModelElement.visibility xmi.value="public"/>
          <Foundation.Core.GeneralizableElement.isRoot xmi.value="true"/>
          <Foundation.Core.GeneralizableElement.isLeaf xmi.value="true"/>
          <Foundation.Core.GeneralizableElement.isAbstract xmi.value="false"/>
          <Foundation.Core.Class.isActive xmi.value="true"/>
          <Foundation.Core.Classifier.feature>
            <Foundation.Core.Attribute xmi.id="i00000003">
              <Foundation.Core.ModelElement.name>
                attribute1
              </Foundation.Core.ModelElement.name>
              <Foundation.Core.ModelElement.visibility xmi.value="public"/>
              <Foundation.Core.Feature.ownerScope xmi.value="instance"/>
              <Foundation.Core.StructuralFeature.multiplicity>
                1
              </Foundation.Core.StructuralFeature.multiplicity>
              <Foundation.Core.StructuralFeature.changeable xmi.value="none"/>
              <Foundation.Core.StructuralFeature.targetScope xmi.value="instance"/>
              <Foundation.Core.Attribute.initialValue>
                <Foundation.Data_Types.Expression>
                  <Foundation.Data_Types.Expression.language>
                  </Foundation.Data_Types.Expression.language>
                  <Foundation.Data_Types.Expression.body>
                    0
                  </Foundation.Data_Types.Expression.body>
                </Foundation.Data_Types.Expression>
              </Foundation.Core.Attribute.initialValue>
              <Foundation.Core.StructuralFeature.type>
                <Foundation.Core.DataType xmi.idref = "i00000004"/> <!-- integer -->
              </Foundation.Core.StructuralFeature.type>
            </Foundation.Core.Attribute>
          </Foundation.Core.Classifier.feature>
        </Foundation.Core.Class>
        <Foundation.Core.DataType xmi.id="i00000004">
          <Foundation.Core.ModelElement.name>
             integer
          </Foundation.Core.ModelElement.name>
          <Foundation.Core.ModelElement.visibility xmi.value="public"/>
```

```
        <Foundation.Core.GeneralizableElement.isRoot xmi.value="true"/>
        <Foundation.Core.GeneralizableElement.isLeaf xmi.value="true"/>
        <Foundation.Core.GeneralizableElement.isAbstract xmi.value="false"/>
      </Foundation.Core.DataType>
    </Foundation.Core.Namespace.ownedElement>
  </Model_Management.Model>
 </XMI.content>
</XMI>
```

C.3 Business Model

The following Model was developed as the source for a simple XML document produced according to XMI.

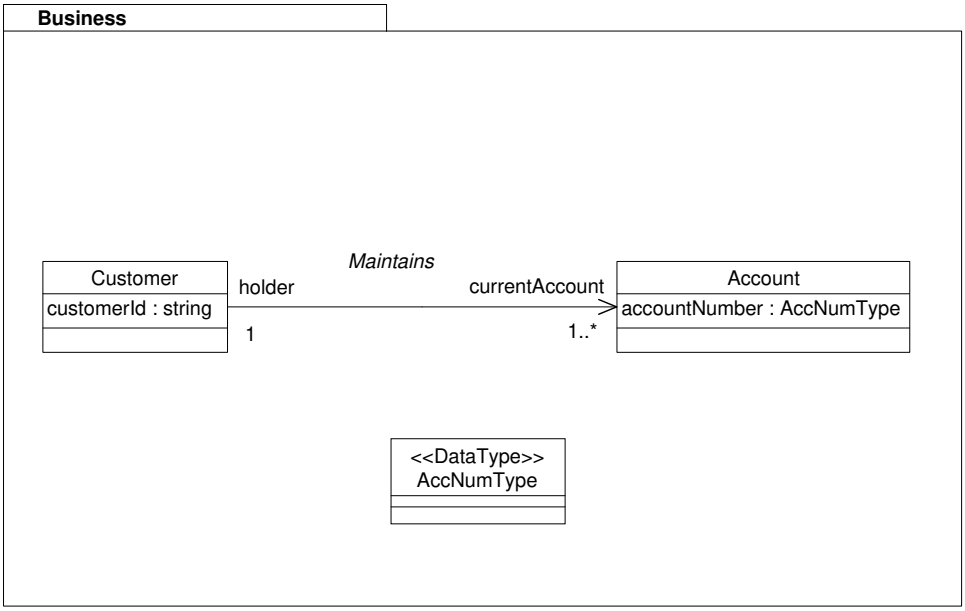

Figure C-2 Simple UML Model Business

The XML resulting from this model is shown below.

```
<?xml version = '1.0' encoding = 'ISO-8859-1' ?>
<!DOCTYPE XMI SYSTEM 'file:uml.dtd' >
<XMI xmi.version = '1.0'>
 <XMI.header>
  <XMI.metamodel xmi.name = 'UML' xmi.version = '1.1'/>
 </XMI.header>
 <XMI.content>
  <Model_Management.Model xmi.id = 'G.1'>
   <Foundation.Core.ModelElement.name>Biz2</Foundation.Core.ModelElement.name>
   <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
   <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'false'/>
```

```
<Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'false'/>
<Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
<Foundation.Core.Namespace.ownedElement>
  <Foundation.Extension_Mechanisms.Stereotype xmi.id = 'G.5'>
    <Foundation.Core.ModelElement.name>
      DataType
    </Foundation.Core.ModelElement.name>
    <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
    <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'false'/>
    <Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'false'/>
    <Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
    <Foundation.Extension_Mechanisms.Stereotype.icon>
      <Foundation.Data_Types.Geometry/>
    </Foundation.Extension_Mechanisms.Stereotype.icon>
    <Foundation.Extension_Mechanisms.Stereotype.baseClass>
      Class
    </Foundation.Extension_Mechanisms.Stereotype.baseClass>
    <Foundation.Extension_Mechanisms.Stereotype.extendedElement>
    </Foundation.Extension_Mechanisms.Stereotype.extendedElement>
  </Foundation.Extension_Mechanisms.Stereotype>
</Foundation.Core.Namespace.ownedElement>
<Foundation.Core.Namespace.ownedElement>
  <Foundation.Core.DataType xmi.id = 'G.7'>
    <Foundation.Core.ModelElement.name>
      String
    </Foundation.Core.ModelElement.name>
    <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
    <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'false'/>
    <Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'false'/>
    <Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
  </Foundation.Core.DataType>
</Foundation.Core.Namespace.ownedElement>
<Foundation.Core.Namespace.ownedElement>
  <Model_Management.Package xmi.id='S.10001'>
    <Foundation.Core.ModelElement.name>
      Business
    </Foundation.Core.ModelElement.name>
    <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
    <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'false'/>
    <Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'false'/>
    <Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
    <Foundation.Core.Namespace.ownedElement>
      <Foundation.Core.Class xmi.id = 'S.10002'>
        <Foundation.Core.ModelElement.name>
          Customer
        </Foundation.Core.ModelElement.name>
        <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
        <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'true'/>
        <Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'true'/>
        <Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
        <Foundation.Core.Class.isActive xmi.value = 'false'/>
        <Foundation.Core.Classifier.feature>
          <Foundation.Core.Attribute xmi.id = 'S.10003'>
            <Foundation.Core.ModelElement.name>
              customerId
```

```
                </Foundation.Core.ModelElement.name>
                <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
                <Foundation.Core.Feature.ownerScope xmi.value = 'instance'/>
                <Foundation.Core.StructuralFeature.multiplicity>
                  1..1
                </Foundation.Core.StructuralFeature.multiplicity>
                <Foundation.Core.StructuralFeature.changeable xmi.value = 'none'/>
                <Foundation.Core.StructuralFeature.targetScope xmi.value =
'instance'/>
                <Foundation.Core.Attribute.initialValue>
                  <Foundation.Data_Types.Expression/>
                </Foundation.Core.Attribute.initialValue>
                <Foundation.Core.StructuralFeature.type>
                  <Foundation.Core.DataType xmi.idref= 'G.7'/> <!-- String -->
                </Foundation.Core.StructuralFeature.type>
              </Foundation.Core.Attribute>
            </Foundation.Core.Classifier.feature>
          </Foundation.Core.Class>
        </Foundation.Core.Namespace.ownedElement>
        <Foundation.Core.Namespace.ownedElement>
          <Foundation.Core.Class xmi.id = 'S.10004'>
            <Foundation.Core.ModelElement.name>
              Account
            </Foundation.Core.ModelElement.name>
            <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
            <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'true'/>
            <Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'true'/>
            <Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
            <Foundation.Core.Class.isActive xmi.value = 'false'/>
            <Foundation.Core.Classifier.associationEnd>
              <Foundation.Core.AssociationEnd xmi.idref= 'G.3'/>
            </Foundation.Core.Classifier.associationEnd>
            <Foundation.Core.Classifier.feature>
              <Foundation.Core.Attribute xmi.id = 'S.10005'>
                <Foundation.Core.ModelElement.name>
                  accountNumber
                </Foundation.Core.ModelElement.name>
                <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
                <Foundation.Core.Feature.ownerScope xmi.value = 'instance'/>
                <Foundation.Core.StructuralFeature.multiplicity>
                  1..1
                </Foundation.Core.StructuralFeature.multiplicity>
                <Foundation.Core.StructuralFeature.changeable xmi.value = 'none'/>
                <Foundation.Core.StructuralFeature.targetScope xmi.value =
'instance'/>
                <Foundation.Core.Attribute.initialValue>
                  <Foundation.Data_Types.Expression/>
                </Foundation.Core.Attribute.initialValue>
                <Foundation.Core.StructuralFeature.type>
                  <Foundation.Core.Class xmi.idref= 'S.10006'/> <!-- AccNumType -->
                </Foundation.Core.StructuralFeature.type>
              </Foundation.Core.Attribute>
            </Foundation.Core.Classifier.feature>
          </Foundation.Core.Class>
        </Foundation.Core.Namespace.ownedElement>
```

```
          <Foundation.Core.Namespace.ownedElement>
            <Foundation.Core.Class xmi.id = 'S.10006'>
              <Foundation.Core.ModelElement.name>
                AccNumType
              </Foundation.Core.ModelElement.name>
              <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
              <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'true'/>
              <Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'true'/>
              <Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
              <Foundation.Core.Class.isActive xmi.value = 'false'/>
              <Foundation.Core.ModelElement.stereotype>
                <Foundation.Extension_Mechanisms.Stereotype xmi.idref= 'G.5'/> <!--
DataType -->
              </Foundation.Core.ModelElement.stereotype>
            </Foundation.Core.Class>
          </Foundation.Core.Namespace.ownedElement>
        </Model_Management.Package>
      </Foundation.Core.Namespace.ownedElement>
      <Foundation.Core.Namespace.ownedElement>
        <Foundation.Core.Association xmi.id = 'G.2'>
          <Foundation.Core.ModelElement.name>
            Maintains
          </Foundation.Core.ModelElement.name>
          <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
          <Foundation.Core.GeneralizableElement.isRoot xmi.value = 'false'/>
          <Foundation.Core.GeneralizableElement.isLeaf xmi.value = 'false'/>
          <Foundation.Core.GeneralizableElement.isAbstract xmi.value = 'false'/>
          <Foundation.Core.Association.connection>
            <Foundation.Core.AssociationEnd xmi.id='G.3'>
              <Foundation.Core.ModelElement.name>
                currentAccount
              </Foundation.Core.ModelElement.name>
              <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
              <Foundation.Core.AssociationEnd.isNavigable xmi.value = 'true'/>
              <Foundation.Core.AssociationEnd.isOrdered xmi.value = 'false'/>
              <Foundation.Core.AssociationEnd.aggregation xmi.value = 'none'/>
              <Foundation.Core.AssociationEnd.multiplicity>
                1..*
              </Foundation.Core.AssociationEnd.multiplicity>
              <Foundation.Core.AssociationEnd.changeable xmi.value = 'none'/>
              <Foundation.Core.AssociationEnd.targetScope xmi.value = 'instance'/>
              <Foundation.Core.AssociationEnd.type>
                <Foundation.Core.Class xmi.idref = 'S.10004'/> <!-- Account -->
              </Foundation.Core.AssociationEnd.type>
            </Foundation.Core.AssociationEnd>
          </Foundation.Core.Association.connection>
          <Foundation.Core.Association.connection>
            <Foundation.Core.AssociationEnd xmi.id = 'G.4'>
              <Foundation.Core.ModelElement.name>
                holder
              </Foundation.Core.ModelElement.name>
              <Foundation.Core.ModelElement.visibility xmi.value = 'public'/>
              <Foundation.Core.AssociationEnd.isNavigable xmi.value = 'false'/>
              <Foundation.Core.AssociationEnd.isOrdered xmi.value = 'false'/>
              <Foundation.Core.AssociationEnd.aggregation xmi.value = 'none'/>
```

```
            <Foundation.Core.AssociationEnd.multiplicity>
              1..1
            </Foundation.Core.AssociationEnd.multiplicity>
            <Foundation.Core.AssociationEnd.changeable xmi.value = 'none'/>
            <Foundation.Core.AssociationEnd.targetScope xmi.value = 'instance'/>
            <Foundation.Core.AssociationEnd.type>
              <Foundation.Core.Class xmi.idref = 'S.10002'/> <!-- Customer -->
            </Foundation.Core.AssociationEnd.type>
          </Foundation.Core.AssociationEnd>
        </Foundation.Core.Association.connection>
      </Foundation.Core.Association>
    </Foundation.Core.Namespace.ownedElement>
  </Model_Management.Model>
 </XMI.content>
</XMI>
```