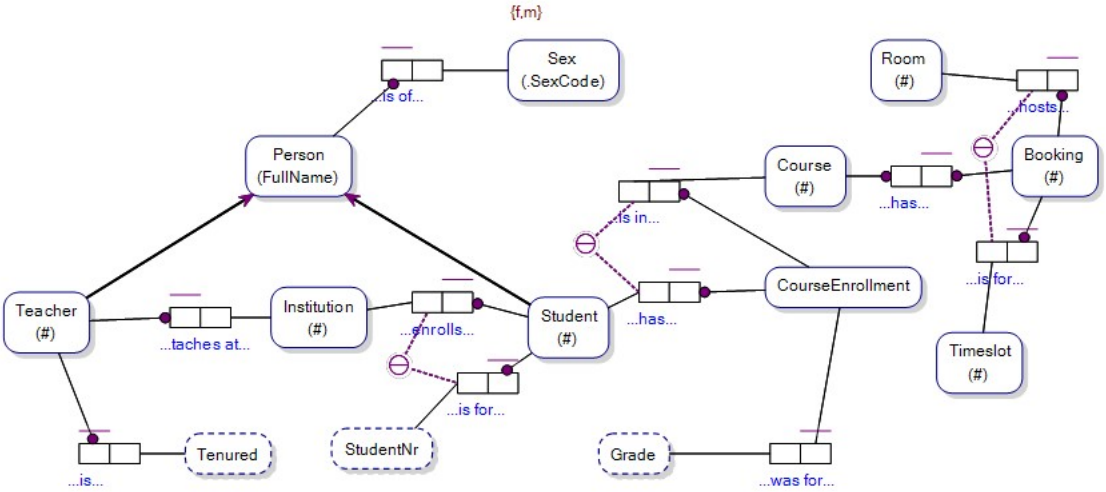
"FactEngine"

Your knowledge engineering partner

Fact Based Modeling – Exchange MetaModel

XML Schema Definition




```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns="" id="Model"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Instance">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" ref="String" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="String" nillable="true">
    <xs:complexType>
      <xs:simpleContent msdata:ColumnName="string_Text" msdata:Ordinal="0">
        <xs:extension base="xs:string" />
      </xs:simpleContent>
    </xs:complexType>
  </xs:element>
```

Version:	V1.1
XSD Version Number:	1.2
Status	Released

Table of Contents

1 About This Document

This document describes the XML Schema Definition (XSD) for XML applications that describe Fact-Based Models.

Fact-Based Modeling includes such modeling techniques as “Object-Role Modeling”.

1.1 Identification

Fact-Based Models conforming to this specification have the following file extension:

File Extension: .fbm

1.1.1 Other extensions

The following are also extensions that are used to identify XML files conforming to this specification:

File Extension: .xml

1.2 XSD Version Nr

This document provides a description of the following XSD Version Number for .fbm XML files:

XSD Version Number: 1.2

2 Instance

Contributors notes here:

2.1 Overview

A Object Type has a set of String values, each of which uniquely identify an Instance of that Object Type.

e.g. An Entity Type may have a unique Instance identified by a GUID.

NB Only Entity Types and Value Types have Instance elements within the exchange metamodel. Instances of Entity Types are instances of a Value Type unless the Entity Type has a Compound Reference Scheme. Fact Types that are 'Objectified' have an Objectifying Entity Type that has unique Instances.

2.2 XSD Text

Below is the XSD text for the Instance artefact:

```
<xs:element name="Instance">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="String" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

3 String

Contributors notes here:

3.1 Overview

A String element is used to store a data value of type, string.

e.g. A string value in a String element may provide the identity of an instance of an Object Type within an Instance element.

See Also: Instance element.

3.2 XSD Text

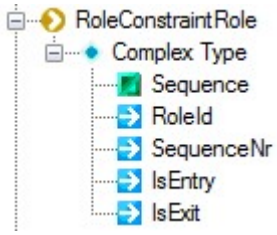
Below is the XSD text for the String artefact:

```
<xs:element name="String" nillable="true">
  <xs:complexType>
    <xs:simpleContent msdata:ColumnName="string_Text" msdata:Ordinal="0">
      <xs:extension base="xs:string" />
    </xs:simpleContent>
  </xs:complexType>
```

3.3 Sample XML

```
<String>c69ea3a6-6d73-47aa-abf6-c52311f4c3a5</String>
```

4 RoleConstraintRole



Contributors notes here:

4.1 Overview

The RoleConstraintRole element stores details of the Role/s referenced by a Role Constraint. Each RoleConstraint element has a set of one or more RoleConstraintRole elements.

Each RoleConstraintRole element has a SequenceNr attribute representing the ordinal number in the sequence of RoleConstraintRole elements for a RoleConstraintRole (e.g. an External Uniqueness Constraint (RoleConstraint) may have a set of two RoleConstraintRole elements each with its own SequenceNr).

4.2 Attribute Definitions – RoleConstraintRole Element

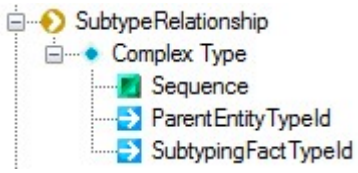
Attribute	Definition
RoleId	The unique identifier of the Role referenced by the RoleConstraintRole element. Represents the Role linked to a RoleConstraint by the RoleConstraintRole.
SequenceNr	The ordinal position of the RoleConstraintRole within the sequence of RoleConstraintRole elements for a RoleConstraint.
IsEntry	Used only for SubsetConstraint, RoleConstraints. Has a value set to ‘true’ is the RoleConstraintRole element represents a RoleConstaint/Role link that is the ‘Entry’ link of a ‘Entry/Exit’ combination of RoleConstraint/Role links, else has a value set to ‘false’. e.g. If a link to a Role from a RoleConstraint is an ‘Entry’ link of a Subset Constraint, the values of Instances referenced by that Role are the subset of Instances referenced by the Role of a corresponding RoleConstraint/Role link that is an ‘Exit’ link.
IsExit	Used only for SubsetConstraint, RoleConstraints. Has a value set to ‘true’ is the RoleConstraintRole element represents a RoleConstaint/Role link that is the ‘Exit’ link of a ‘Entry/Exit’ combination of RoleConstraint/Role links, else has a value set to ‘false’.

4.3 XSD Text

Below is the XSD text for the RoleConstraintRole artefact:

```
<xs:element name="RoleConstraintRole">
  <xs:complexType>
    <xs:sequence />
    <xs:attribute name="RoleId" type="xs:string" />
    <xs:attribute name="SequenceNr" type="xs:string" />
    <xs:attribute name="IsEntry" type="xs:string" />
    <xs:attribute name="IsExit" type="xs:string" />
  </xs:complexType>
</xs:element>
```

5 SubtypeRelationship



Contributors notes here:

5.1 Overview

The Fact element stores the set of FactData relevant to a Fact of a FactType.

5.2 Attribute Definitions – SubtypeRelationship Element

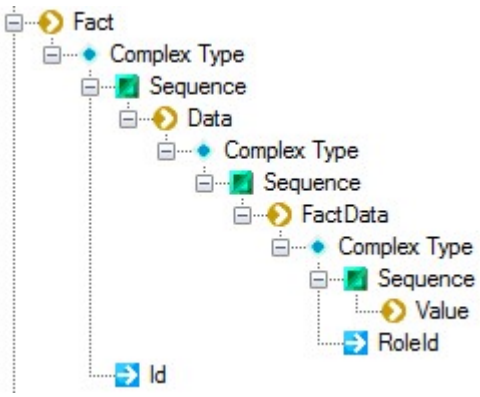
Attribute	Definition
ParentEntityTypeId	The unique identifier of the EntityType that is the SuperType of the EntityType within which the SubtypeRelationship artefact exists. **NB** The value of a ParentEntityTypeId attribute may be the unique identifier of an Objectifying Entity Type (of an Objectified Fact Type)
SubtypingFactTypeId	The unique identifier of the FactType that provides the identity relationship between a Subtype and a SuperType. **NB** SubtypingFactTypes are not generally visible to people viewing a Fact-Based Model. In an ORM Diagram, the link between an Entity Type that is a Subtype of another Object Type (Entity Type or Objectified Fact Type) is displayed as an arrow directed line.

5.3 XSD Text

Below is the XSD text for the SubtypeRelationship artefact:

```
<xs:element name="SubtypeRelationship">
  <xs:complexType>
    <xs:sequence />
    <xs:attribute name="ParentEntityTypeId" type="xs:string" />
    <xs:attribute name="SubtypingFactTypeId" type="xs:string" />
  </xs:complexType>
</xs:element>
```

6 Fact



Contributors notes here:

6.1 Overview

The Fact element stores the set of FactData relevant to a Fact of a FactType.

6.2 Attribute Definitions – Fact Element

Attribute	Definition
Id	The unique identifier of the Fact represented by the Fact element.

6.3 Attribute Definitions – FactData Element

Attribute	Definition
RoleId	Has a value equal to the unique identifier of the Role within the Fact Type of the Fact for which the Fact Data is an element.

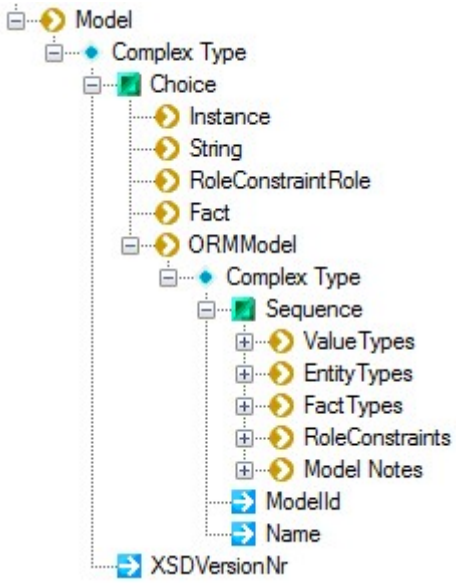
6.4 XSD Text

Below is the XSD text for the Instance artefact:

```
<xs:element name="Fact">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Data">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="FactData">
              <xs:complexType>
                <xs:sequence>
                  <xs:element msdata:Ordinal="1" minOccurs="0" name="Value" type="xs:string" />
                </xs:sequence>
                <xs:attribute name="RoleId" type="xs:string" />
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
```

```
      </xs:element>
    </xs:sequence>
    <xs:attribute name="Id" type="xs:string" />
   </xs:complexType>
</xs:element>
```

7 Model



Contributors notes here:

7.1 Overview

The 'Model' element contains all elements that go into describing a Fact-Based Model.

7.2 Attribute Definitions – Model Element

Attribute	Definition
XSDVersionNr	The version of this XML Schema under which the XML application was generated.

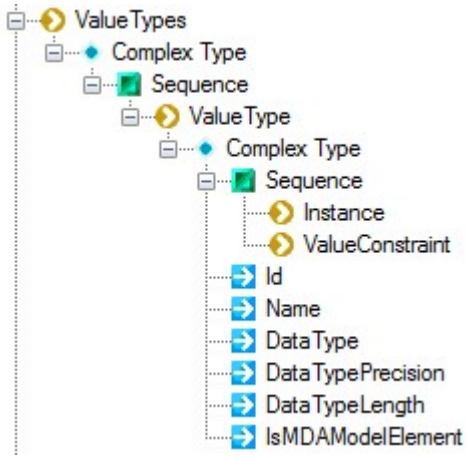
7.3 Attribute Definitions – ORMModel Element

Attribute	Definition
ModelId	The unique identifier for the Model.
Name	The name of the Model.

7.4 XSD Text

No XSD text is provided for the Model artefact here.
See Appendix A

8 Value Types / ValueType / ValueConstraint



Contributors notes here:

8.1 Overview

8.1.1 ValueTypes Element

The ValueTypes element contains a list of ValueType elements. Each ValueType element describes a Value Type within a Fact-Based Model.

8.1.2 ValueType Element

The ‘ValueType’ element describes a Value Type within a Fact-Based Model.

8.1.3 ValueConstraint Element

The ValueConstraint element contains a sequence of String.

8.2 Attribute Definitions – ValueType Element

Attribute	Definition
Id	The unique identifier of the ValueType represented by the ValueType element.
Name	The name of the ValueType represented by the ValueType element.
DataType	The data-type of the ValueType represented by the ValueType element.
DataTypePrecision	If the ‘DataType’ of the Value Type requires a degree of precision, this attribute stores that degree of precision.
DataTypeLength	If the ‘DataType’ of the Value Type requires a degree of length, this attribute stores that degree of length. e.g. A ‘Fixed-Length Text’ data type requires the length of the data

	values for a Value Type of that data type.
IsMDAModelElement	Set to 'true' if the Value Type is for the MetaModel of a language other than a Fact-Based Modeling language, else set to 'false'. This attribute is used when using this specification for the exchange of 'Model Driven Architecture' models in languages other than a FBM language. Non-FBM languages MetaModels and FBM Models may exist within the one XML Application. **Notes:** Fact-Based Modeling MetaModels may be used to exchange languages other than Fact-Based Modeling languages (e.g. ORM Models) such as ER Diagrams, UML Class Diagrams, Flowcharts, Data Flow Diagrams etc. If a FBM MetaModel is used to exchange languages other than Fact-Based Modeling languages, the MetaModel of that language (e.g. the ER Diagram metamodel) is stored as a logical injection inside the same MetaModel as FBM Models. That is, the MetaModel of other languages may be exchanged using this specification. The MetaModel of another language, and the corresponding Models, may be exchanged together with a Fact-Based Model within the same MetaModel. i.e. An instance of an XML application corresponding to this XSD specification may contain both FBM Models and Models of language other than FBM within the one XML document. Those Model Elements of a Model of a language other than FBM that are stored within an XML application conforming to this XSD specification are flagged with this attribute set to 'True'. This avoids confusion when reviewing or importing Models from within an XML application conforming to this specification. i.e. e.g. Software importing solely Fact-Based Models from an XML document conforming to this specification may ignore those Model Elements that are flagged as 'MDA Model Elements' when displaying a list of the Model Elements (e.g. Value Types) within the primary Fact-Based Model exchanged using the standard.

8.3 XSD Text

Below is the XSD text for the ValueType element:

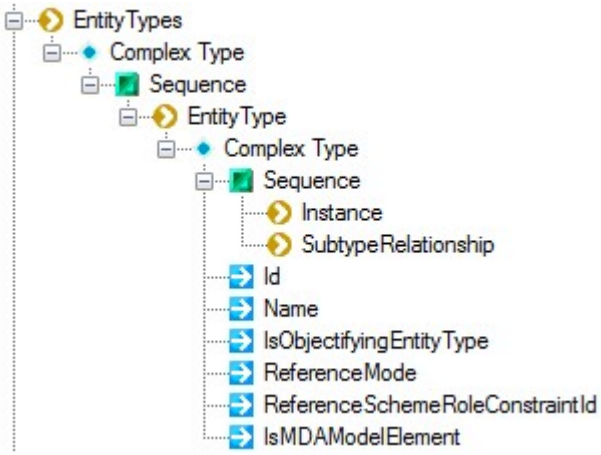
```
<xs:element minOccurs="0" maxOccurs="unbounded" name="ValueTypes">
  <xs:complexType>
    <xs:sequence>
```

```
      <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueType">
       <xs:complexType>
        <xs:sequence>
         <xs:element minOccurs="0" maxOccurs="unbounded" ref="Instance" />
         <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueConstraint" />
        </xs:sequence>
        <xs:attribute name="Id" type="xs:string" />
        <xs:attribute name="Name" />
        <xs:attribute name="DataType" type="xs:string" />
        <xs:attribute name="DataTypePrecision" type="xs:string" />
        <xs:attribute name="DataTypeLength" type="xs:string" />
        <xs:attribute name="IsMDAModelElement" />
       </xs:complexType>
      </xs:element>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
```

8.3.1 Sample XML

None provided with this release.

9 EntityTypes / EntityType



Contributors notes here:

9.1 Overview

9.1.1 EntityTypes Element

The EntityTypes element contains a sequence of EntityType elements. Each EntityType element represents an EntityType within a Fact-Based Model.

9.1.2 EntityType Element

The EntityType element stores the details of an Entity Type within a Fact-Based Model.

9.2 Attribute Definitions – EntityType Element

Attribute	Definition
Id	The unique identifier of the Entity Type within the Fact-Based Model. **NB** Is unique amongst the Id values for: Value Types; Entity Types; Fact Types; Role Constraints; Model Notes;
Name	The name of the Entity Type. **NB** Is unique amongst the Id values for: Value Types; Entity Types; Fact Types; Role Constraints;
IsObjectifyingEntityType	If the EntityType is the objectifying Entity Type of an

	Objectified Fact Type then this attribute has a value of 'true', else this attribute has a value of 'false'.
ReferenceMode	The string value of the ReferenceMode of the Entity Type if the Entity Type does not have a Compound Reference Scheme, but has a Reference Mode, otherwise an empty string. If the Reference Mode for an Entity Type is not yet defined, then this attribute contains an empty string.
ReferenceShemeRoleConstraintId	If the EntityType defined within an EntityType element has a Reference Scheme, contains a reference to the RoleConstraint that ranges over the Reference Scheme of the Entity Type.
IsMDAModelElement	Set to 'true' if the Entity Type is for the MetaModel of a language other than a Fact-Based Modeling language, else set to 'false'. This attribute is used when using this specification for the exchange of 'Model Driven Architecture' models in languages other than a FBM language. Non-FBM languages MetaModels and FBM Models may exist within the one XML application. **Notes:** Fact-Based Modeling MetaModels may be used to exchange languages other than Fact-Based Modeling languages (e.g. ORM Models) such as ER Diagrams, UML Class Diagrams, Flowcharts, Data Flow Diagrams etc. If a FBM MetaModel is used to exchange languages other than Fact-Based Modeling languages, the MetaModel of that language (e.g. the ER Diagram metamodel) is stored as a logical injection inside the same MetaModel as FBM Models. That is, the MetaModel of other languages may be exchanged using this specification. The MetaModel of another language, and the corresponding Models, may be exchanged together with a Fact-Based Model within the same MetaModel. i.e. An instance of an XML application corresponding to this XSD specification may contain both FBM Models and Models of language other than FBM within the one XML document. Those Model Elements of a Model of a language other than FBM that are stored within an XML application conforming to this XSD specification are flagged with

	this attribute set to ‘True’. This avoids confusion when reviewing or importing Models from within an XML application conforming to this specification. i.e. e.g. Software importing solely Fact-Based Models from an XML document conforming to this specification may ignore those Model Elements that are flagged as ‘MDA Model Elements’ when displaying a list of the Model Elements (e.g. Value Types) within the primary Fact-Based Model exchanged using the standard.

9.3 Attribute Definitions – ReferenceSchemeRoleConstraint

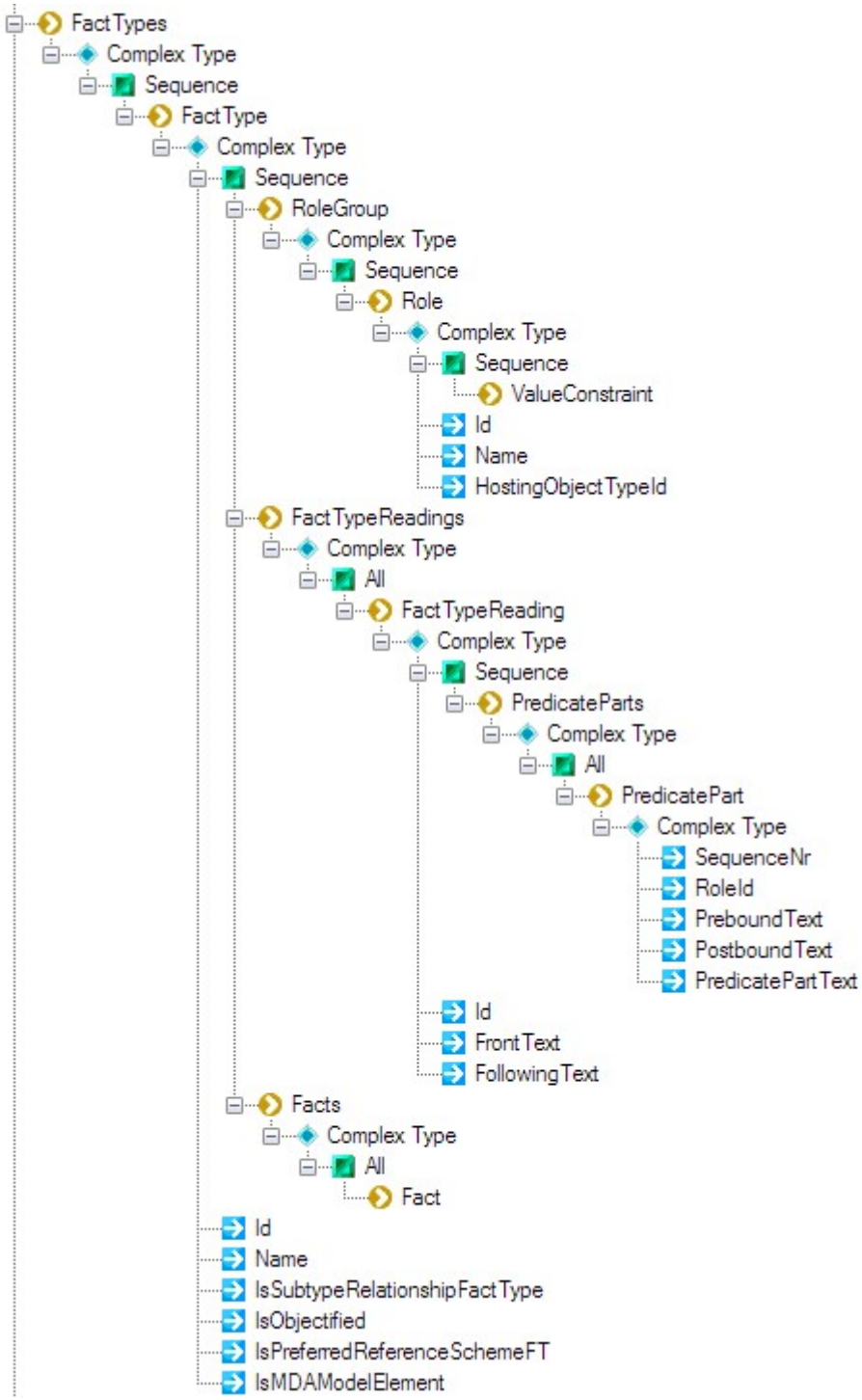
Attribute	Definition
Id	The unique identifier of the Role Constraint that ranges over the Reference Scheme for the Entity Type.

9.4 XSD Text

Below is the XSD text for the EntityTypes artefact:

```
<xs:element minOccurs="0" maxOccurs="unbounded" name="EntityTypes">
 <xs:complexType>
  <xs:sequence>
   <xs:element minOccurs="0" maxOccurs="unbounded" name="EntityType">
    <xs:complexType>
     <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="Instance" />
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="SubtypeRelationship" />
     </xs:sequence>
     <xs:attribute name="Id" type="xs:string" />
     <xs:attribute name="Name" type="xs:string" />
     <xs:attribute name="IsObjectifyingEntityType" type="xs:string" />
     <xs:attribute name="ReferenceMode" />
     <xs:attribute name="ReferenceModeRoleConstraintId" type="xs:string" />
     <xs:attribute name="IsMDAModelElement" />
    </xs:complexType>
   </xs:element>
  </xs:sequence>
 </xs:complexType>
</xs:element>
```

10 Fact Types / Fact Type Elements



Contributors notes here:

10.1 Overview

The "FactTypes" Element contains a list of the Elements containing details of the Fact Types within the Model.

10.2 Attribute Definitions – FactType Element

Attribute	Definition
Id	The unique identifier of the FactType within the Model.
Name	The name of the FactType within the Model.
IsSubtypeConstraintFactType	Has a value set to 'true' if the FactType represents the identity Fact Type of an Entity Type, else has a value set to 'false'. NB If an Entity Type is a subtype of another Entity Type, then a Fact Type (not displayed on the model) called an 'Identity Fact Type' is implied by the Subtype Constraint link between the subtype Entity Type and the supertype Entity Type. This attribute allows easy identification of those Fact Types which are Identity Fact Types and that are not displayed on the Model.
IsObjectified	Has a value set to 'true' if the Fact Type is 'Objectified', else has a value set to 'false'.
IsPreferredReferenceSchemeFT	Has a value set to 'true' if the Fact Type links an Entity Type to a Value Type and where that Entity Type does not have a Compound Reference Scheme, else has a value set to 'false'. Fact Types where this attribute has a value set to 'true' are not displayed in the Model under the standard view of an Entity Type/Reference Mode relationship (i.e. Where the Entity Type has a Simple Reference Scheme), but may be displayed if the software allows the user to view the relationship between the Entity Type and Value Type of the Reference Mode. If an Entity Type has a Reference Scheme that is not a Compound Reference Scheme, then the Fact Type mapping the Entity Type to the Value Type of the Reference Scheme has this attribute having a value set to 'true'. Fact Types that are in no way associated with the Reference Scheme of an Entity Type always have this attribute having a value set to 'false'.
IsMDAModelElement	Set to 'true' if the Fact Type is for the MetaModel of a language other than a Fact-Based Modeling language, else set to 'false'.

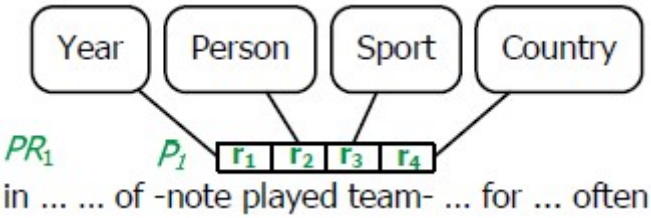
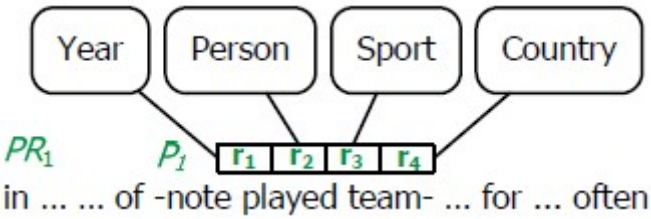
	This attribute is used when using this specification for the exchange of 'Model Driven Architecture' models in languages other than a FBM language. Non-FBM languages MetaModels and FBM Models may exist within the one XML Application. **Notes:** Fact-Based Modeling MetaModels may be used to exchange languages other than Fact-Based Modeling languages (e.g. ORM Models) such as ER Diagrams, UML Class Diagrams, Flowcharts, Data Flow Diagrams etc. If a FBM MetaModel is used to exchange languages other than Fact-Based Modeling languages, the MetaModel of that language (e.g. the ER Diagram metamodel) is stored as a logical injection inside the same MetaModel as FBM Models. That is, the MetaModel of other languages may be exchanged using this specification. The MetaModel of another language, and the corresponding Models, may be exchanged together with a Fact-Based Model within the same MetaModel. i.e. An instance of an XML application corresponding to this XSD specification may contain both FBM Models and Models of language other than FBM within the one XML document. Those Model Elements of a Model of a language other than FBM that are stored within an XML application conforming to this XSD specification are flagged with this attribute set to 'True'. This avoids confusion when reviewing or importing Models from within an XML application conforming to this specification. i.e. e.g. Software importing solely Fact-Based Models from an XML document conforming to this specification may ignore those Model Elements that are flagged as 'MDA Model Elements' when displaying a list of the Model Elements (e.g. Value Types) within the primary Fact-Based Model exchanged using the standard.

10.3 Attribute Definitions – Role Element

Attribute	Definition
Id	The unique identifier of the Role within the Model.
Name	If the Role has a Role Name, then this attribute has a value equal to that Role Name, otherwise this attribute has a blank value.

JoinedObjectTypeId	The unique identifier of the Object Type linked by the Role within the Model.

10.4 Attribute Definitions – Fact Type Reading Element

Attribute	Definition
Id	The unique identifier of the Fact Type Reading. Most be unique within the whole Model.
FrontText	e.g. “in” as within the Fact Type Reading outlined in the ORM Diagram below and as within the Fact Type Reading: “in Year Person of –note played team-Sport for Country often”  e.g. “tickets for” as in the Fact Type Readiing: “tickets for Booking are being mailed to Address”
FollowingText	e.g. “often” as within the Fact Type Reading outlined in the ORM Diagram below and as within the Fact Type Reading: “in Year Person of –note played team-Sport for Country often” 

10.5 Attribute Definitions – Predicate Part Element

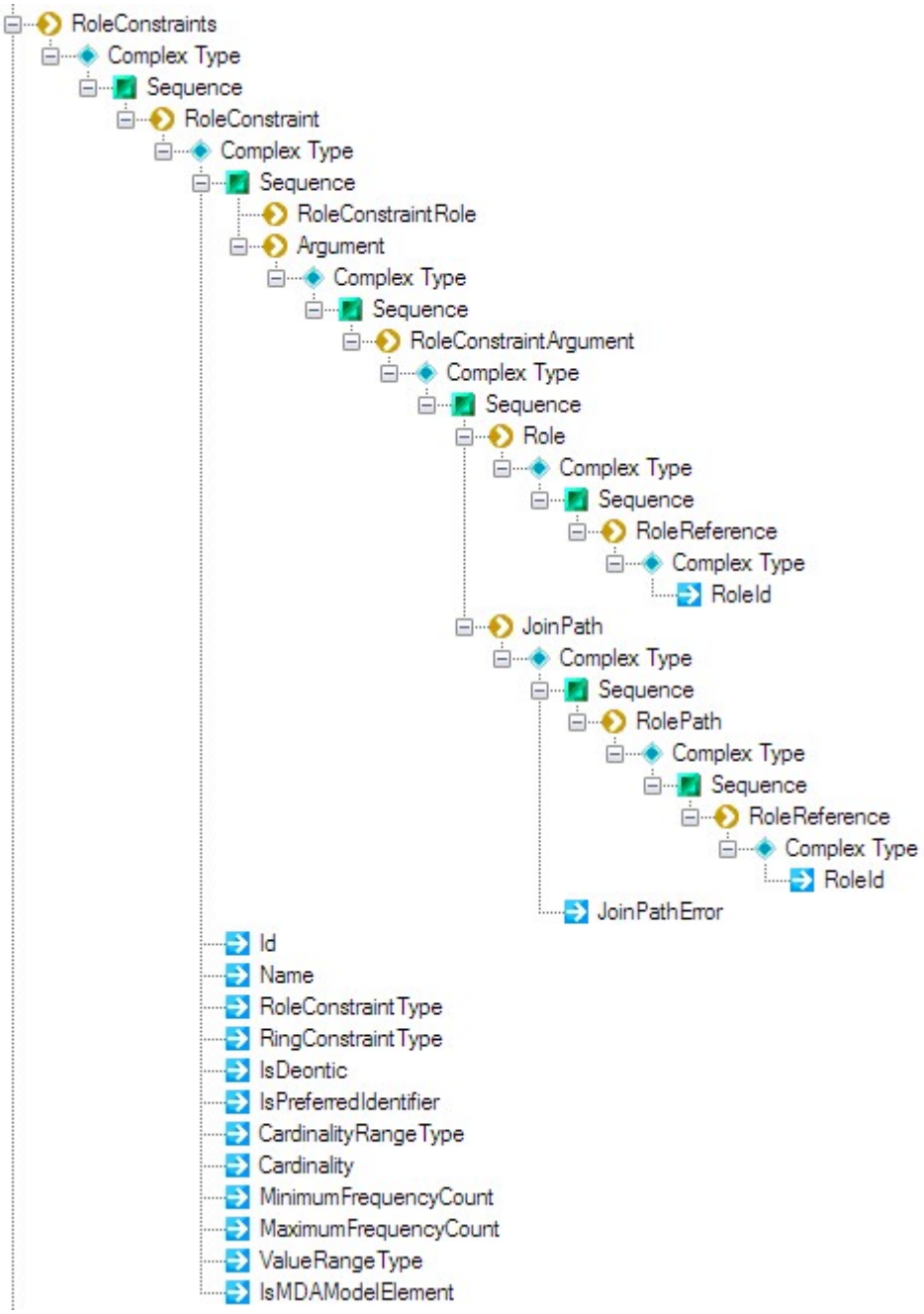
Attribute	Definition
SequenceNr	Integer representing the ordinal position of the Predicate Part within the sequence of Predicate Parts within the Fact Type Reading.
RoleId	The unique Id of the Role to which the Predicate Part is assigned within the Fact Type Reading.
PreboundText	e.g. “first” as within the Fact Type Reading “Person has first-Name” and where “has” is the PredicatePartText of the first Predicate Part (SequenceNr =1) of the Fact Type Reading.
PostboundText	e.g.
PredicatePartText	e.g. “has” as within the Fact Type Reading, “Person has first-Name” and where “has” is the PredicatePartText of the first Predicate Part within the Fact Type Reading.

10.6 XSD Text

Below is the XSD text for the FactType artefact:

```
<xs:element minOccurs="0" maxOccurs="unbounded" name="FactType">
 <xs:complexType>
  <xs:sequence>
   <xs:element minOccurs="0" maxOccurs="unbounded" name="RoleGroup">
    <xs:complexType>
     <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Role">
       <xs:complexType>
        <xs:sequence>
         <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueConstraint" />
        </xs:sequence>
        <xs:attribute name="Id" type="xs:string" />
        <xs:attribute name="Name" type="xs:string" />
        <xs:attribute name="HostingObjectTypeId" type="xs:string" />
       </xs:complexType>
      </xs:element>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
   <xs:element name="FactTypeReadings">
    <xs:complexType name="New_ComplexType">
     <xs:all>
      <xs:element name="FactTypeReading">
       <xs:complexType name="New_ComplexType">
        <xs:sequence>
         <xs:element name="PredicateParts">
          <xs:complexType name="New_ComplexType">
           <xs:all>
            <xs:element name="PredicatePart">
             <xs:complexType name="New_ComplexType">
              <xs:attribute name="SequenceNr" />
              <xs:attribute name="RoleId" />
              <xs:attribute name="PreboundText" />
              <xs:attribute name="PostboundText" />
              <xs:attribute name="PredicatePartText" />
             </xs:complexType>
            </xs:element>
           </xs:all>
          </xs:complexType>
         </xs:element>
        </xs:sequence>
        <xs:attribute name="Id" />
        <xs:attribute name="FrontText" />
       </xs:complexType>
      </xs:element>
     </xs:all>
    </xs:complexType>
   </xs:element>
   <xs:element name="Facts">
    <xs:complexType name="New_ComplexType">
     <xs:all>
      <xs:element minOccurs="0" maxOccurs="1" ref="Fact" />
     </xs:all>
    </xs:complexType>
   </xs:element>
  </xs:sequence>
  <xs:attribute name="Id" type="xs:string" />
  <xs:attribute name="Name" type="xs:string" />
  <xs:attribute name="IsSubtypeRelationshipFactType" type="xs:string" />
  <xs:attribute name="IsObjectified" type="xs:string" />
  <xs:attribute name="IsPreferredReferenceSchemeFT" type="xs:string" />
  <xs:attribute name="IsMDAModelElement" />
 </xs:complexType>
</xs:element>
```

11 Role Constraint



Contributors notes here:

11.1 Overview

The RoleConstraints element provides a sequence of RoleConstraint elements, with each RoleConstraint element containing the details of a RoleConstraint within the Fact-Based Model.

11.2 Attribute Definitions – Role Constraint Element

Attribute	Definition
Id	The unique identifier of the Role Constraint within the Model.
Name	The name of the Role Constraint.
RoleConstraintType	The type of Role Constraint represented by the element.
RingConstraintType	If the RoleConstraintType attribute of the Role Constraint element has a value of 'RingConstraint' then this attribute has a value of the type of Ring Constraint.
Cardinality	
IsDeontic	Has a value set to 'true' if the Role Constraint is 'Deontic', else has a value set to 'false'.

CardinalityRangeType	If the RoleConstraint is a 'Cardinality Constraint' or a 'Frequency Constraint', this attribute stores the cardinality range of the Role Constraint.
	Values exclusively include:
	`LessThanOREqual`
	`Equal`
	`GreaterThanOREqual`
	`Between`

MinimumFrequencyCount	If the RoleConsrtaintType attribute has a value of 'FrequencyConstraint' then this attribute has a value set to the minimum in the range of the Frequency Constraint.
MaximumFrequencyCount	If the RoleConsrtaintType attribute has a value of 'FrequencyConstraint' then this attribute has a value set to the maximum in the range of the Frequency Constraint.
IsPreferredIdentifier	Has a value set to 'true' if the Role Constraint represented by the element represents the 'Preferred Identification Scheme' for an Entity Type.

ValueRangeType	If the RoleConstraint is a 'Value-Comparison Constraint' (i.e. has a 'RoleConstraintType' value of 'ValueComparisonConstraint', this attribute stores the type of Value-Comparison constrained by the Role Constraint.
	Values exclusively include:
	`LessThan`
	`LessThanOrEqual`
	`GreaterThan`
	`GreaterThanOREqual`

IsMDAModelElement	Set to 'true' if the Role Constraint is for the MetaModel of a language other than a Fact-Based Modeling language, else set to 'false'. This attribute is used when using this specification for the exchange of 'Model Driven Architecture' models in languages other than a FBM language. Non-FBM languages MetaModels and FBM Models may exist within the one XML Application.

	Notes: Fact-Based Modeling MetaModels may be used to exchange languages other than Fact-Based Modeling languages (e.g. ORM Models) such as ER Diagrams, UML Class Diagrams, Flowcharts, Data Flow Diagrams etc. If a FBM MetaModel is used to exchange languages other than Fact-Based Modeling languages, the MetaModel of that language (e.g. the ER Diagram metamodel) is stored as a logical injection inside the same MetaModel as FBM Models. That is, the MetaModel of other languages may be exchanged using this specification. The MetaModel of another language, and the corresponding Models, may be exchanged together with a Fact-Based Model within the same MetaModel. i.e. An instance of an XML application corresponding to this XSD specification may contain both FBM Models and Models of language other than FBM within the one XML document. Those Model Elements of a Model of a language other than FBM that are stored within an XML application conforming to this XSD specification are flagged with this attribute set to 'True'. This avoids confusion when reviewing or importing Models from within an XML application conforming to this specification. i.e. e.g. Software importing solely Fact-Based Models from an XML document conforming to this specification may ignore those Model Elements that are flagged as 'MDA Model Elements' when displaying a list of the Model Elements (e.g. Value Types) within the primary Fact-Based Model exchanged using the standard.

11.3 Element Definition – RoleConstraintRole

Each RoleConstraint element has one or more RoleConstraintRole elements.

The RoleConstraintRole element defines the relationship between a RoleConstraint and the Roles to which the RoleConstraint applies.

See the definition of the RoleConstraintRole element for more details.

11.4 Element Definition - Argument

The Argument element has a set of RoleConstraintArgument elements. Each RoleConstraintArgument element has a Role and JoinPath element.

The Role element contains a set of RoleReference elements which contain details of the Roles that make up the Role Constraint Argument.

11.5 Element Definition – Join Path

The JoinPath element contains the RolePath element which contains the ordered sequence of Roles that make up the Join Path between the Roles of the associated Role Constraint Argument.

11.5.1 Attribute Definitions – RoleReference Element

Attribute	Definition
JoinPathError	Stores the error code, if an error exists in the construction of the JoinPath for the associated RoleConstraintArgument

11.5.2 Attribute Definitions – RoleReference Element

Attribute	Definition
RoleId	The unique identifier of the Role within the Argument

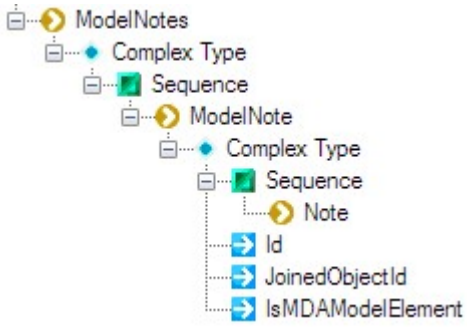
11.6 XSD Text

Below is the XSD text for the Role Constraint artefact:

```
<xs:element minOccurs="0" maxOccurs="unbounded" name="RoleConstraints">
 <xs:complexType>
  <xs:sequence>
   <xs:element minOccurs="0" maxOccurs="unbounded" name="RoleConstraint">
    <xs:complexType>
     <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" ref="RoleConstraintRole" />
      <xs:element name="Argument">
       <xs:complexType name="New_ComplexType">
        <xs:sequence>
         <xs:element name="RoleConstraintArgument">
          <xs:complexType name="New_ComplexType">
           <xs:sequence>
            <xs:element name="Role">
             <xs:complexType name="New_ComplexType">
              <xs:sequence>
               <xs:element name="RoleReference">
                <xs:complexType>
                 <xs:attribute name="RoleId" />
                </xs:complexType>
               </xs:element>
              </xs:sequence>
             </xs:complexType>
            </xs:element>
            <xs:element name="JoinPath">
             <xs:complexType name="New_ComplexType">
```

```
                    <xs:sequence>
                      <xs:element name="RolePath">
                        <xs:complexType name="New_ComplexType">
                          <xs:sequence>
                            <xs:element name="RoleReference">
                              <xs:complexType>
                                <xs:attribute name="RoleId" />
                              </xs:complexType>
                            </xs:element>
                          </xs:sequence>
                        </xs:complexType>
                      </xs:element>
                    </xs:sequence>
                    <xs:attribute name="JoinPathError" />
                  </xs:complexType>
                </xs:element>
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:sequence>
        <xs:attribute name="Id" type="xs:string" />
        <xs:attribute name="Name" type="xs:string" />
        <xs:attribute name="RoleConstraintType" type="xs:string" />
        <xs:attribute name="RingConstraintType" type="xs:string" />
        <xs:attribute name="IsDeontic" type="xs:string" />
        <xs:attribute name="IsPreferredIdentifier" type="xs:string" />
        <xs:attribute name="CardinalityRangeType" type="xs:string" />
        <xs:attribute name="Cardinality" type="xs:string" />
        <xs:attribute name="MinimumFrequencyCount" type="xs:string" />
        <xs:attribute name="MaximumFrequencyCount" type="xs:string" />
        <xs:attribute name="ValueRangeType" />
        <xs:attribute name="IsMDAModelElement" />
      </xs:complexType>
    </xs:element>
  </xs:sequence>
 </xs:complexType>
</xs:element>
```

12 Model Note



Contributors notes here:

12.1 Overview

The ModelNotes element contains a list of ModelNote elements, with each ModelNote element providing the details of a Model Note within the Fact-Based Model.

Attribute	Definition
Id	The unique identifier of the Role Constraint within the Model.
IsMDAModelElement	Set to 'true' if the Model Note is for the MetaModel of a language other than a Fact-Based Modeling language, else set to 'false'. This attribute is used when using this specification for the exchange of 'Model Driven Architecture' models in languages other than a FBM language. Non-FBM languages MetaModels and FBM Models may exist within the one XML Application. **Notes:** Fact-Based Modeling MetaModels may be used to exchange languages other than Fact-Based Modeling languages (e.g. ORM Models) such as ER Diagrams, UML Class Diagrams, Flowcharts, Data Flow Diagrams etc. If a FBM MetaModel is used to exchange languages other than Fact-Based Modeling languages, the MetaModel of that language (e.g. the ER Diagram metamodel) is stored as a logical injection inside the same MetaModel as FBM Models. That is, the MetaModel of other languages may be exchanged using this specification. The MetaModel of another language, and the corresponding Models, may be exchanged together with a Fact-Based Model within the same MetaModel. i.e. An instance of an XML application corresponding to this XSD specification may contain both FBM Models and Models of language other than FBM within the one XML document.

	Those Model Elements of a Model of a language other than FBM that are stored within an XML application conforming to this XSD specification are flagged with this attribute set to 'True'. This avoids confusion when reviewing or importing Models from within an XML application conforming to this specification. i.e. e.g. Software importing solely Fact-Based Models from an XML document conforming to this specification may ignore those Model Elements that are flagged as 'MDA Model Elements' when displaying a list of the Model Elements (e.g. Value Types) within the primary Fact-Based Model exchanged using the standard.

12.2 XSD Text

Below is the XSD text for the Instance artefact:

```
<xs:element minOccurs="0" maxOccurs="unbounded" name="Model Notes">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="Model Note">
        <xs:complexType>
          <xs:sequence>
            <xs:element msdata:Ordinal="1" minOccurs="0" name="Note" type="xs:string" />
          </xs:sequence>
          <xs:attribute name="Id" type="xs:string" />
          <xs:attribute name="JoinedObjectId" type="xs:string" />
          <xs:attribute name="IsMDAModelElement" />
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:sequence>
```

13 Appendix A – XSD For the ‘Model’ Artefact

```
<xs:element name="ORMModel">
<xs:complexType>
 <xs:sequence>
  <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueTypes">
   <xs:complexType>
    <xs:sequence>
     <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueType">
      <xs:complexType>
       <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" ref="Instance" />
        <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueConstraint" />
       </xs:sequence>
       <xs:attribute name="Id" type="xs:string" />
       <xs:attribute name="Name" />
       <xs:attribute name="DataType" type="xs:string" />
       <xs:attribute name="DataTypePrecision" type="xs:string" />
       <xs:attribute name="DataTypeLength" type="xs:string" />
       <xs:attribute name="IsMDAModelElement" />
      </xs:complexType>
     </xs:element>
    </xs:sequence>
   </xs:complexType>
  </xs:element>
  <xs:element minOccurs="0" maxOccurs="unbounded" name="EntityTypes">
   <xs:complexType>
    <xs:sequence>
     <xs:element minOccurs="0" maxOccurs="unbounded" name="EntityType">
      <xs:complexType>
       <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" ref="Instance" />
        <xs:element minOccurs="0" maxOccurs="unbounded" ref="SubtypeRelationship" />
       </xs:sequence>
       <xs:attribute name="Id" type="xs:string" />
       <xs:attribute name="Name" type="xs:string" />
       <xs:attribute name="IsObjectifyingEntityType" type="xs:string" />
       <xs:attribute name="ReferenceMode" />
       <xs:attribute name="ReferenceSchemeRoleConstraintId" type="xs:string" />
       <xs:attribute name="IsMDAModelElement" />
      </xs:complexType>
     </xs:element>
    </xs:sequence>
   </xs:complexType>
  </xs:element>
  <xs:element minOccurs="0" maxOccurs="unbounded" name="FactTypes">
   <xs:complexType>
    <xs:sequence>
     <xs:element minOccurs="0" maxOccurs="unbounded" name="FactType">
      <xs:complexType>
       <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="RoleGroup">
         <xs:complexType>
          <xs:sequence>
           <xs:element minOccurs="0" maxOccurs="unbounded" name="Role">
            <xs:complexType>
             <xs:sequence>
              <xs:element minOccurs="0" maxOccurs="unbounded" name="ValueConstraint" />
             </xs:sequence>
             <xs:attribute name="Id" type="xs:string" />
             <xs:attribute name="Name" type="xs:string" />
             <xs:attribute name="JoinedObjectTypeId" type="xs:string" />
            </xs:complexType>
           </xs:element>
          </xs:sequence>
         </xs:complexType>
        </xs:element>
        <xs:element minOccurs="0" maxOccurs="unbounded" ref="Fact" />
       </xs:sequence>
```

```
        <xs:attribute name="Id" type="xs:string" />
        <xs:attribute name="Name" type="xs:string" />
        <xs:attribute name="IsSubtypeRelationshipFactType" type="xs:string" />
        <xs:attribute name="IsObjectified" type="xs:string" />
        <xs:attribute name="IsPreferredReferenceSchemeFT" type="xs:string" />
        <xs:attribute name="IsMDAModelElement" />
       </xs:complexType>
      </xs:element>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
   <xs:element minOccurs="0" maxOccurs="unbounded" name="RoleConstraints">
    <xs:complexType>
     <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="RoleConstraint">
       <xs:complexType>
        <xs:sequence>
         <xs:element minOccurs="0" maxOccurs="unbounded" ref="RoleConstraintRole" />
         <xs:element name="SetComparisonConstraint">
          <xs:complexType">
           <xs:sequence>
            <xs:element name="SetComparisonArgument">
             <xs:complexType ">
              <xs:sequence>
               <xs:element name="RolePosition">
                <xs:complexType ">
                 <xs:sequence>
                  <xs:element name="CompatibleRole">
                   <xs:complexType ">
                    <xs:attribute name="RoleId" />
                   </xs:complexType>
                  </xs:element>
                  <xs:element name="ComparableRole">
                   <xs:complexType ">
                    <xs:attribute name="RoleId" />
                   </xs:complexType>
                  </xs:element>
                 </xs:sequence>
                 <xs:attribute name="RoleId" />
                 <xs:attribute name="PositionId" />
                </xs:complexType>
               </xs:element>
              </xs:sequence>
              <xs:attribute name="Arity" />
             </xs:complexType>
            </xs:element>
           </xs:sequence>
           <xs:attribute name="ArgumentLength" />
           <xs:attribute name="NrArguments" />
          </xs:complexType>
         </xs:element>
        </xs:sequence>
        <xs:attribute name="Id" type="xs:string" />
        <xs:attribute name="Name" type="xs:string" />
        <xs:attribute name="RoleConstraintType" type="xs:string" />
        <xs:attribute name="RingConstraintType" type="xs:string" />
        <xs:attribute name="IsDeontic" type="xs:string" />
        <xs:attribute name="IsPreferredIdentifier" type="xs:string" />
        <xs:attribute name="CardinalityRangeType" type="xs:string" />
        <xs:attribute name="Cardinality" type="xs:string" />
        <xs:attribute name="MinimumFrequencyCount" type="xs:string" />
        <xs:attribute name="MaximumFrequencyCount" type="xs:string" />
        <xs:attribute name="ValueRangeType" />
        <xs:attribute name="IsMDAModelElement" />
       </xs:complexType>
      </xs:element>
     </xs:sequence>
    </xs:complexType>
   </xs:element>
   <xs:element minOccurs="0" maxOccurs="unbounded" name="ModelNotes">
    <xs:complexType>
     <xs:sequence>
```

```
            <xs:element minOccurs="0" maxOccurs="unbounded" name="ModelNote">
              <xs:complexType>
                <xs:sequence>
                  <xs:element msdata:Ordinal="1" minOccurs="0" name="Note" type="xs:string" />
                </xs:sequence>
                <xs:attribute name="Id" type="xs:string" />
                <xs:attribute name="JoinedObjectId" type="xs:string" />
                <xs:attribute name="IsMDAModelElement" />
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
    <xs:attribute name="ModelId" type="xs:string" />
    <xs:attribute name="Name" type="xs:string" />
  </xs:complexType>
</xs:element>
```

14 Appendix B - Outstanding ORM2 constructs

The following lists the constructs within ORM2 that are not yet implemented by this schema.

ORM2 Construct	Description
Role Value Constraints	Role Value Constraints constrain the data values that may be present in sample populations (Facts) within the FactData element, and for a particular Role.

15 Reviewers Notes and Issues

Open Issues

Issue #	Raised By	Description
20150428-05	CH	How do you handle a disjunctive mandatory constraint? Note that these are different from Frequency constraints with a minimum of 1. 20150428-VM-I don't know what a Disjunctive Mandatory Constraint is. Can you create one in NORMA and dump an image here? Can you provide a précis of what one is here? 20150510-VM-Clifford to comment. 20150511-CH-(Comments received from CH via email) Here's a disjunctive mandatory constraint in CQL: "either Employee participates in company-Superannuation or that Employee has personal-Superannuation;" I.e. can be both, but must be one. There's a four-way DMC in this diagram, which says that EventDate must have some content: <dataconstellation.com/ActiveFacts/examples/images/Genealogy.png> Note that this is also an example of disjunctive identification, which is a two-edged sword. 20150512-VM-How is this handled by the FBMWG and/or an example from NORMA?

Closed Issues

This section is empty. Reviewers requiring the list of closed issues should refer to the dropbox.com Dropbox for this specification and view the in-progress documents.

16 Revision History

The below is the revision history for this document.

Revision	Notes
V0.7	- Added SubtypeRelationship element definition - Changed the name of Attribute 'IsSubtypeConstraintFactType' to 'IsSubtypeRelationshipFactType' in FactType element.
V0.8	- Added DataTypePrecision and DataTypeLength to the ValueType element.
V0.81	- Change 'JoinedObjectId' to 'JoinedObjectTypeId' in Role element - Added ValueConstraint element to Role element's complex type/sequence
V0.82	- Updated Issues Register. Will close items in next version.
V0.83	- Changed all references to 'Compound Reference Mode' to 'Compound Reference Scheme'. - Modified 'ReferenceMode' to 'ReferenceScheme' where required. See closed issues. - Added 'IsMDAModelElement' attribute to each of the ValueType, EntityType, FactType, RoleConstraint and ModelNote elements. - See dates of Closed Issue Items for those closed with this version release. Dated, 20150510 through 20150512
V1.0	- Initial release of this specification.
V1.1	- Fixed errata in the document. EntityType element had a sub element, "ReferenceSchemeRoleConstraint", listed within the text. Rather, "ReferenceSchemeRoleConstraintId" is an attribute of the EntityType element. - Concatenated "Mode Notes" to "ModelNotes" and "Model Note" to "ModeNote" within the respective elements. - Removed "name="New_ComplexType" on complextype sections where this was not required. Was artefact introduced by XSD editor.

End of Document