



Batch Markup Language

BatchML- Batch Production Record Version 0500 – March 2011



IMPORTANT: While the information, data, and standards provided in this publication were developed and are presented in good faith in accordance with a reasonable process that was subject to intellectual property and antitrust policies to benefit the industry as a whole, the publication is provided "as is" for information and guidance only, and there is no representation or warranty of any type or kind, including but not limited to warranties of merchantability or fitness for a particular purpose, and no warranty that use of the information, data, or standards will not infringe patent, copyright, trademark, trade secret, or other intellectual property rights of any party.

Table of Contents

1	Schema Scope	4
1.1	Referenced Schemas.....	4
1.2	Key Use Assumptions	4
1.3	Key Information Assumptions	4
1.4	Common Data Types	4
1.5	Core Components	4
2	Schema Organization.....	5
2.1	BatchProductionRecord Element.....	5
2.2	Type Names	5
2.3	User Element Extensibility	5
3	Batch production record Models.....	6
3.1	Batch production record	6
3.2	Records with references to other records	7
3.3	Event model	8
3.4	Sample model	8
3.5	Data set model	9
3.6	Master recipe, control recipes, and recipe procedure elements	9
3.7	Production request, production response, product definition elements	10
4	Element Definitions	11
4.1	Transactions.....	11
4.2	BatchProductionRecordEntryType Group.....	12
4.3	Batch Production Record Element.....	14
4.4	Data Elements.....	22
5	Diagram Convention.....	38

Change History:

Change	Date	Person	Description
V0500	Mar 2011	Dennis Brandl	Initial version Includes elements from ANSI/ISA 95.02-2010 for Operations Performance, Operations Schedule, Operations Definitions, and Physical Asset resources

Copyright © 2010 WBF The Organization for Production Technology
All Rights Reserved. <http://www.wbf.org>

This WBF (including specifications, documents, software, and related items) referred to as the Batch Markup Language (BatchML) is provided by the copyright holders under the following license.

Permission to use, copy, modify, or redistribute this Work and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted provided the WBF is acknowledged as the originator of this Work using the following statement:

"The Batch Markup Language (BatchML) is used courtesy of WBF."

In no event shall WBF, its members, or any third party be liable for any costs, expenses, losses, damages or injuries incurred by use of the Work or as a result of this agreement.

Material from ANSI/ISA-88 and ANSI/ISA-95 series of standards used with permission of ISA - The Instrumentation, Systems, and Automation Society, www.isa.org

1 Schema Scope

This document provides explanatory information about the referenced WBF XML schemas used to exchange information about batch production records, called the Batch Markup Language, or BatchML.

This information is based on the data models and attributes defined in the ANSI/ISA 88.00.04 Batch Production Record Information. Contact ISA (The Instrumentation, Systems, and Automation Society) for copies of the standard. Additional information on the standard is available at www.isa.org.

1.1 Referenced Schemas

This document provides addresses the contents of the following WBF XML schema:

BatchML-V0500-BatchProductionRecord.xsd

1.2 Key Use Assumptions

The schemas define exchanged information and do not define the use of the information or encapsulation of the information in any defining transactions. These schemas are intended to be used to create XML documents used to exchange batch data as well as serve as the basis for corporate, system or application specific schemas that may be derived from the BatchML schemas.

1.3 Key Information Assumptions

The schemas define simple and complex types and elements for recipe, equipment and batch list data commonly found in batch applications. A set of data models is presented for recipes, equipment and batch lists. Each model also illustrates the equivalent top-level XML elements that correspond to top-level objects identified in the ANSI/ISA-88 standard. The details of the schema element and attribute definitions are contained in later sections of this document.

1.4 Common Data Types

The BatchML BatchInformation schema used the B2MML Common schema to pick up common data types. See the documentation for the Common Types in the file:

B2MML-V0500-Common.doc

1.5 Core Components

The BatchML BatchInformation schema used the B2MML Core Component schemas.

The base types for most elements are derived from core component types that are compatible with the UN/CEFACT core component types. The UN/CEFACT core component types are a common set of types that define specific terms with semantic meaning (e.g. the meaning of a quantity, currency, amount, identifier,...). The UN/CEFACT core components were defined in a Core Components Technical Specification (CCTS) developed by the ebXML project now organized by UN/CEFACT and ISO TC 154.

The core components are defined in the schema file:

B2MML-V0500-CoreComponents.xsd

2 Schema Organization

2.1 BatchProductionRecord Element

The BatchML root element is BatchProductionRecord.

2.2 Type Names

The XML schema uses a model that defines simple and complex data types for each element. The data types all follow the convention of a suffix of “Type” added to the element name.

Schema definition:

```
<xsd:element name = "ApprovalDate" type = "ApprovalDateType" ==  
  
<xsd:simpleType name="ApprovalDateType">  
    <xsd:restriction base="xsd:dateTime">  
    </xsd:restriction>  
</xsd:simpleType>
```

2.3 User Element Extensibility

In order to make the schemas more useful, they include the ability for elements to be extended. The extended elements are not defined in this standard and should not be considered understandable between applications without prior agreement.

See the definition of user extensions in:

B2MML-V0500-Extensions.doc

3 Batch production record Models

The exchanged information is derived from the UML models below.

NOTE: This version has used the ANSI/ISA 95.02-2010 definition for the ISA 95 data elements. The data definitions in ANSI/ISA 88.04 have been extended to include the following:

1. Addition of the OperationsDefinition schema definition.
2. Addition of the OperationsPerformance schema definition.
3. Addition of the OperationsSchedule schema definition.
4. Addition of PhysicalAssetIDs where ever EquipmentIDs are defined.
5. Addition of ResourceUse and ResourceType elements in a ResourceQualificationManifest to match the ISA 95 resource use and resource type definitions.

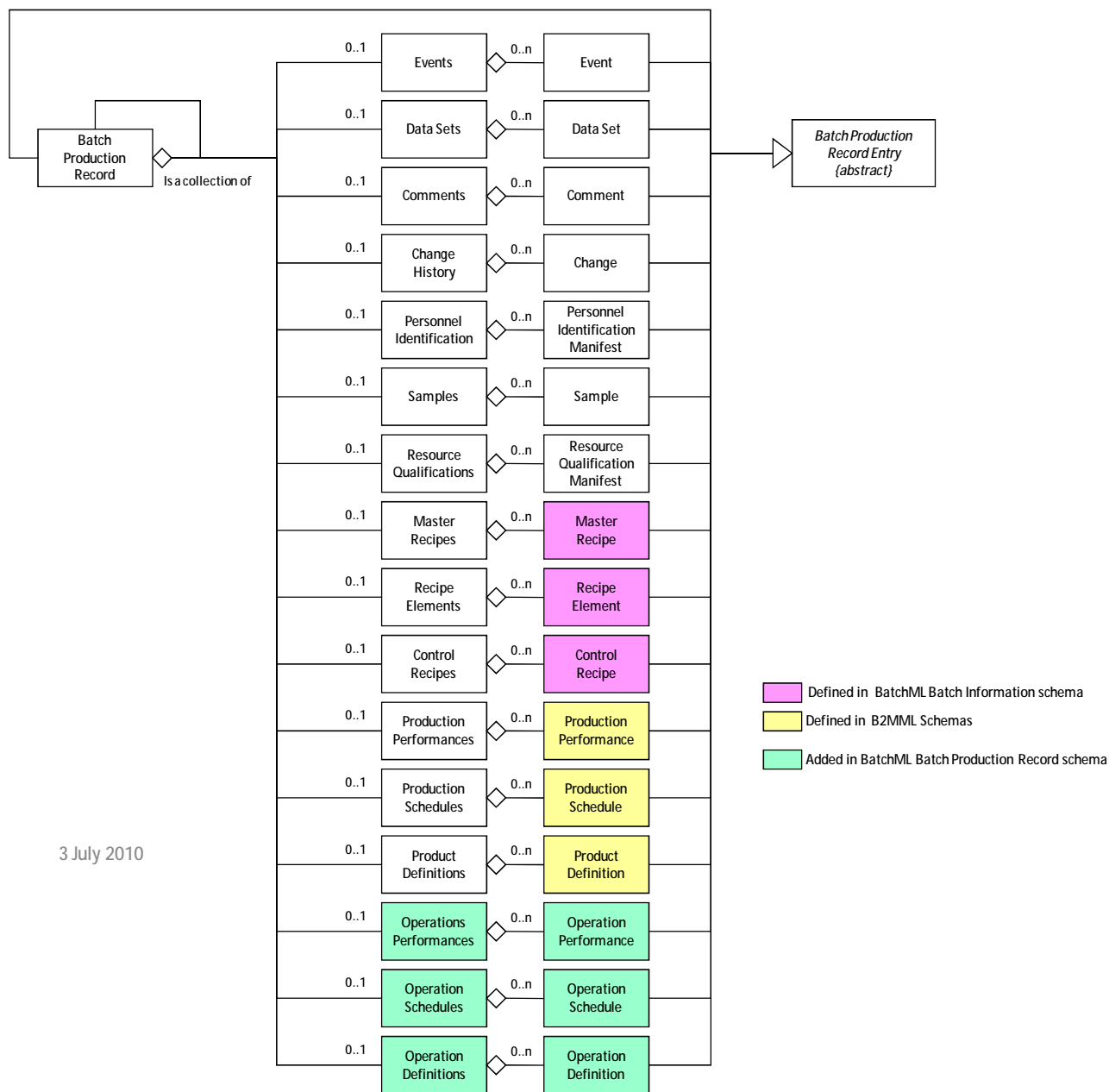
3.1 Batch production record

A batch production records is made up of multiple sub-records acting as container objects. These are represented as XML elements. Each sub-record is a collection of similarly typed individual records.

A Batch Production Record may also contain another Batch Production Record.

Figure 1 is the UML model derived from ANSI/ISA-88.04 and ANSI/ISA95.02-2010. Each of the sub-record container elements contains a collection of zero or more individual records. These are modeled in BatchML as XML elements within the sub-record elements.

All of the individual records are specialized types derived from a Batch Production Record Entry abstract type. This is modeled in BatchML using the "group" **BatchProductionRecordEntryType** element that is included as the initial element in each of the individual records.



3.2 Records with references to other records

The following four individual records may reference other records; comments, personnel identification manifest, qualification manifest, and change history. These are modeled in UML as in Figure 2. These are modeled as a **RecordReference** element, which is defined as a Core Component **IdentifierType**. This should contain the EntryID value of the referenced value. Figure 2 is the UML model from ANSI/ISA-88.04.

Example 1: The EntryID contains an XPath specification string that identifies the element in the XML record.

Example 2: The EntryID contains a unique number that for each element.

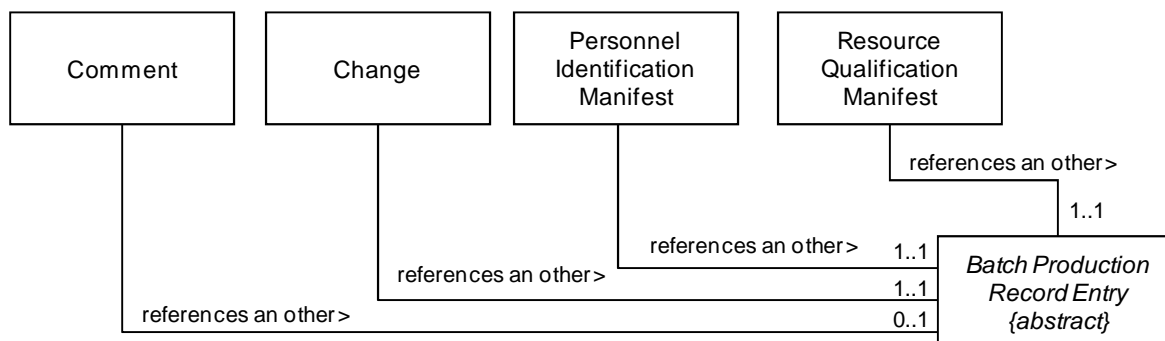


Figure 2 - Record with references to other records

3.3 Event model

An event is a discrete occurrence in time. There are different types of event objects and each event is identified by an event type and subtype. Some event objects may be associated with other events, these are modeled as a **RecordReference** element, which is defined as a Core Component **IdentifierType**. This should contain an XPath specification string that identifies the element in the XML record.

Events contain user defined records, may contain alarm event information and a set of event associations. User defined attributes are also derived from the Batch Production Record Entry type. Figure 3 is the UML model from ANSI/ISA-88.04.

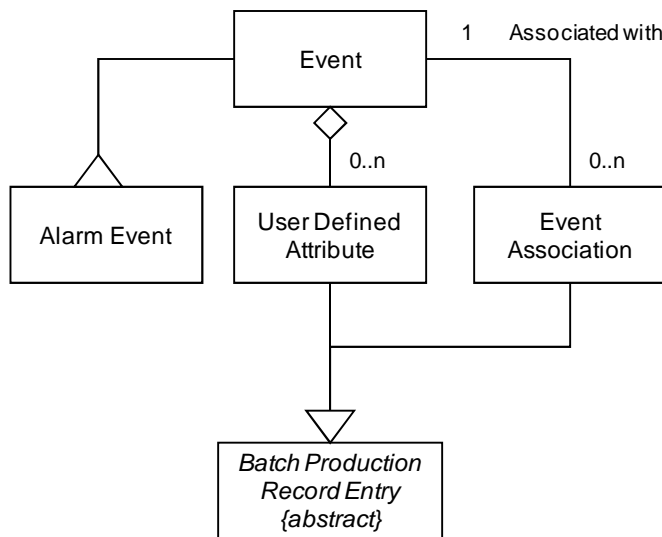


Figure 3 - Event model

3.4 Sample model

Samples contain Sample Tests, and Sample Tests contain Sample Test Results. Sample Tests and Sample Test Results are also derived from the Batch Production Record Entry type. Figure 4 is the UML model from ANSI/ISA-88.04.

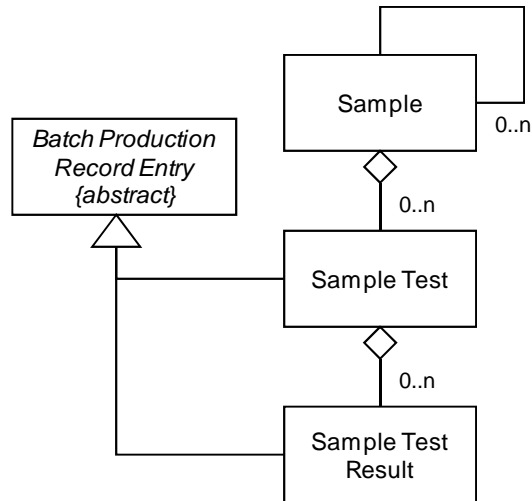


Figure 4 - Sample model

3.5 Data set model

Data sets are used to represent multivariable data sets or time based data sets. The sets contain ordered data sets. Figure 5 is the UML model from ANSI/ISA-88.04. The ordered data set is modeled using a "choice" element as either:

1. A Delimited data block in which the data sets are recorded in a single delimited string, with the delimiting characters defined in the **DelimitedDataBlockType**.
2. As a set of **OrderedDataType** elements. Each **OrderedDataType** contains an optional time element and a set of data elements.

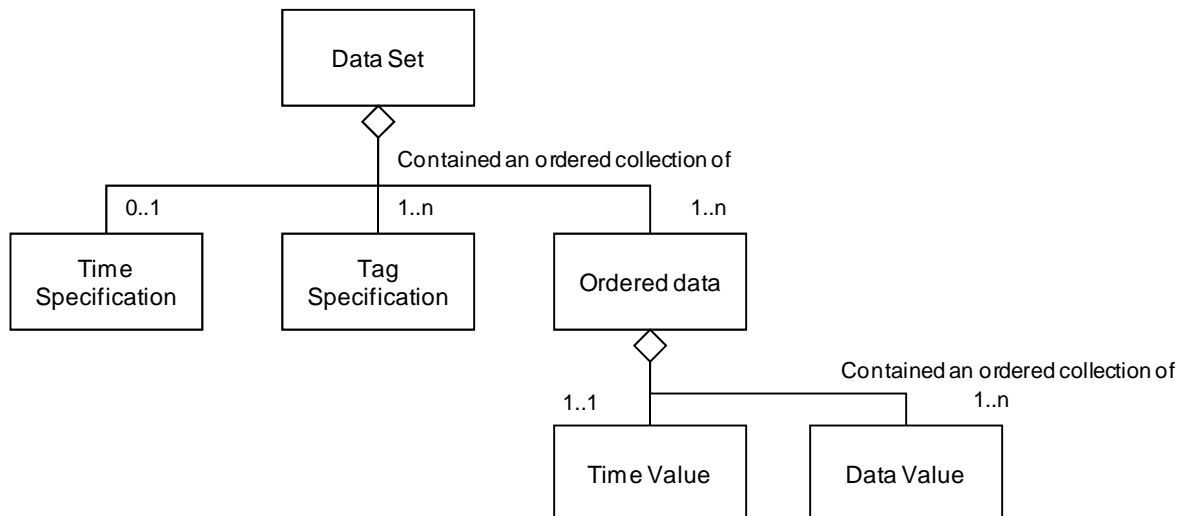


Figure 5 - Data sets

3.6 Master recipe, control recipes, and recipe procedure elements

The master recipe, control recipe, and recipe procedure elements are defined in ANSI/ISA-88.01 standard. The Batch Production Record schema uses the element definitions defined in:

BatchML-V0500-BatchInformation.doc

The Batch Production Record schema uses the schema definitions from the BatchML file:

BatchML-V0500-BatchInformation.xsd

3.7 Production request, production response, product definition elements

The production request, production response, and product definition elements are defined in the ANSI/ISA-95.02 standard. The Batch Production Record schema uses the element definitions defined in:

B2MML-V0500-ProductDefinition.doc
B2MML-V0500-ProductionPerformance.doc
B2MML-V0500-ProductionSchedule.doc

The Batch Production Record schema uses the schema definitions from the B2MML files:

B2MML-V0500-ProductDefinition.xsd
B2MML-V0500-ProductionPerformance.xsd
B2MML-V0500-ProductionSchedule.xsd

4 Element Definitions

The term BatchProductionRecord is shorted to **BPR** in the descriptions.

4.1 Transactions

The ANSI/ISA88-04 standard does not define transactions for Batch Production Records. BatchML defines a set of standard transactions based on the ISA-95.05 Business to Manufacturing Transaction standards. See ANSI/ISA95.05 or IEC 62264-5 for further details on transactions.

The following top level elements for transactions are defined:

Transaction Element	Description
GetBatchProductionRecord	<p>Defines a message used to implement a GET action on a BPR.</p> <ul style="list-style-type: none"> • BPR ID specified: Defines a request that the receiver is to return, in a SHOW message, all attributes and contained elements of the BPR that matches the ID for the specified equipment scope. • Wildcard BPR ID specified: Defines a request that the receiver is to return, in a SHOW message, all attributes and contained elements of all BPRs that match the ID wildcard for the specified equipment scope. • No BPR ID Specified: Apply the additional constraints. • One or more of the following constraints may apply: <ul style="list-style-type: none"> ○ CreationDate specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified creation date for the specified equipment scope. ○ BatchProductionRecordSpec specified: Defines a request that the receiver is to return, in a SHOW message, BPRs created with the BatchProductionRecordSpec for the specified equipment scope. ○ BatchID specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified BatchID for the specified equipment scope. ○ CampaignID specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified CampaignID for the specified equipment scope. ○ ExpirationDate specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified ExpirationDate for the specified equipment scope. ○ LastChangedDate specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified LastChangedDate for the specified equipment scope. ○ LotID specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified LotID for the specified equipment scope. ○ MaterialDefinitionID specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified MaterialDefinitionID for the specified equipment scope. ○ RecordStatus specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified RecordStatus for the specified equipment scope.

	<ul style="list-style-type: none"> ○ Version specified: Defines a request that the receiver is to return, in a SHOW message, BPRs with the specified Version for the specified equipment scope.
ShowBatchProductionRecord	Response from a GetBatchProductionRecord transaction.
ProcessBatchProductionRecord	Defines a request that the receiver is to add a new BPR. Any assigned IDs in the BPR are returned in the AcknowledgeBatchProductionRecord message.
AcknowledgeBatchProductionRecord	Response to a ProcessBatchProductionRecord transaction.
ChangeBatchProductionRecord	Defines a request to change information in a BPR, replacing the existing information with the information in the transaction message.
RespondBatchProductionRecord	Response to the ChangeBatchProductionRecord transaction.
CancelBatchProductionRecord	Defines a request to cancel a BPR.
SyncBatchProductionRecord	Defines a received message that may contain a new, changed, or deleted BPR.

4.2 BatchProductionRecordEntryType Group

BatchProductionRecordEntryType is a group definition that is included in most of the other BatchProductionRecord elements.

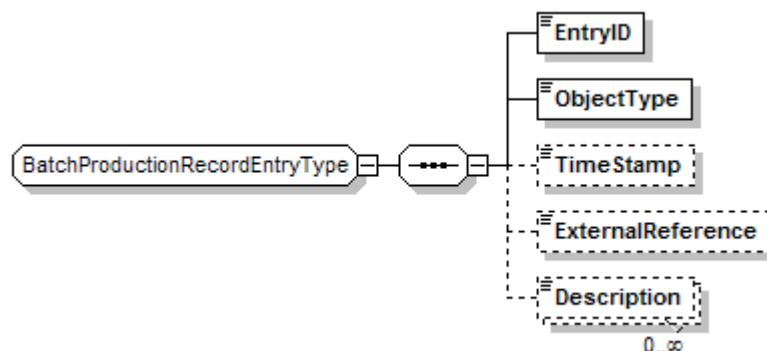


Figure 6 - Batch Production Record Entry Type - Group Definition

Element/Type	Description
Entry ID IdentifierType	A unique identification of the BPR individual element. This element is mandatory.
ObjectType RecordObjectTypeType	<p>Identifies the type of object an entry is based upon. This element is mandatory. This may be either a standard type or an application specific extended type. Standard enumerations correspond to the BPR element types and are:</p> <ul style="list-style-type: none"> • Batch Production Record • Change • Comment • Control Recipe • Data Set • Event • Master Recipe • Operations Definition • Operations Performance

	<ul style="list-style-type: none"> • Operations Schedule • Product Definition • Production Performance • Production Schedule • Personnel Identification Manifest • Resource Definition Manifest • Recipe Element • Sample • Sample Test • Sample Test Result • Other <p>If "Other" then the type is an application specific extension and the value is defined in the attribute "OtherValue".</p>
TimeStamp DateTimeType	The optional time stamp associated with the entry.
ExternalReference IdentifierType	Contains an optional reference to data which is stored external to the BPR.
Description DescriptionType	Additional optional additional information about the BPR.

4.3 Batch Production Record Element

A top level element is a BatchProductionRecord element or type BatchProductionRecordType.

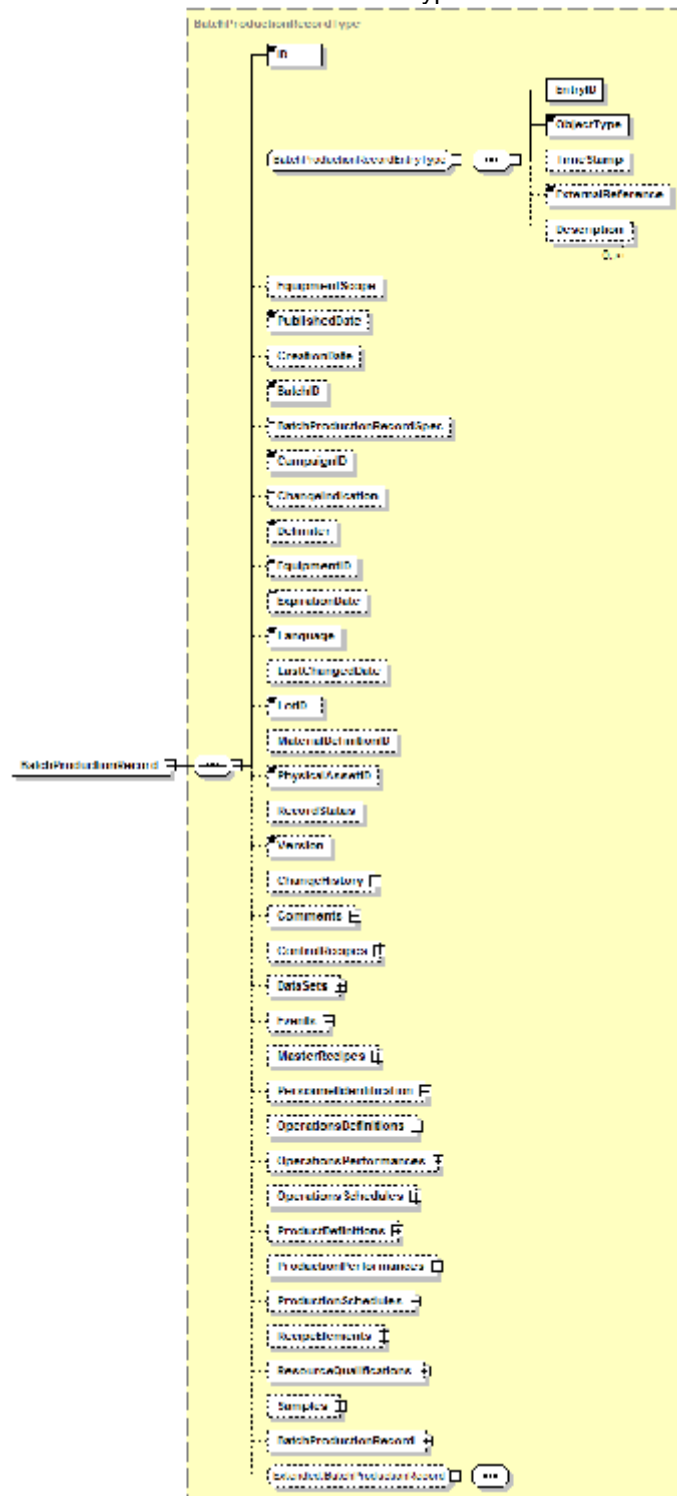
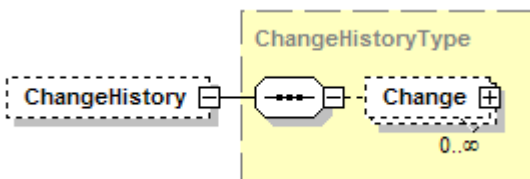
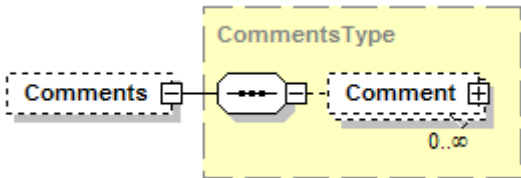
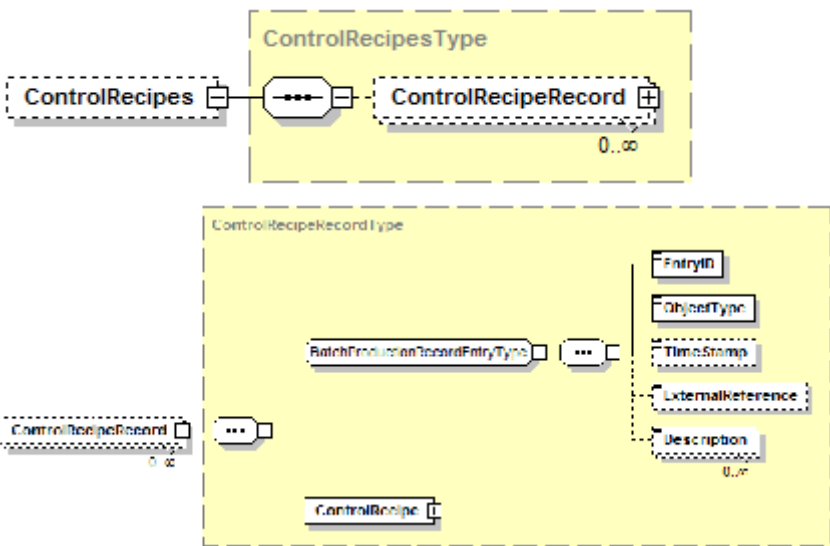
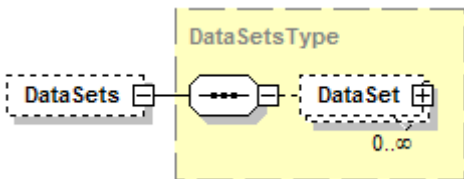
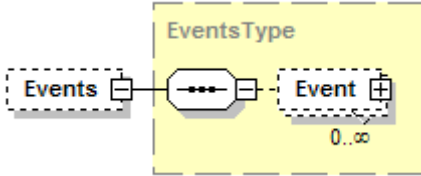
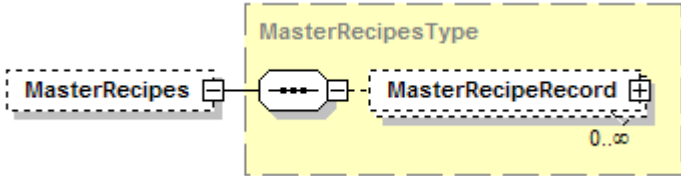
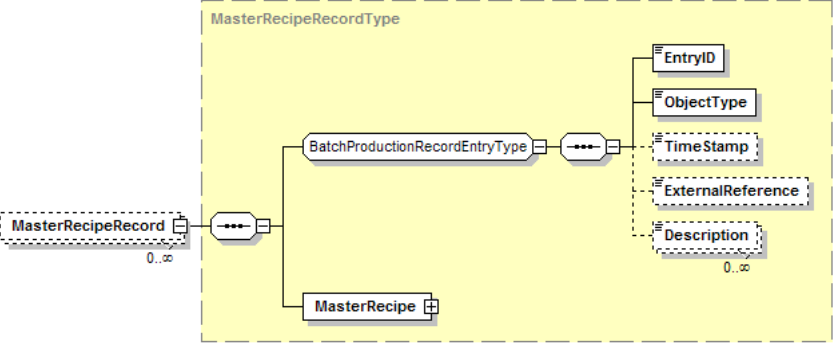
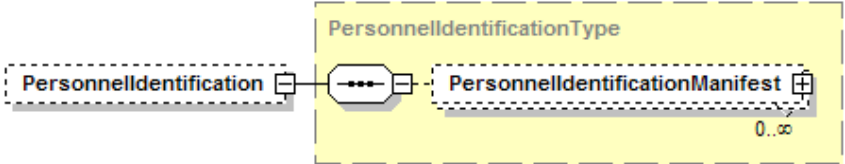
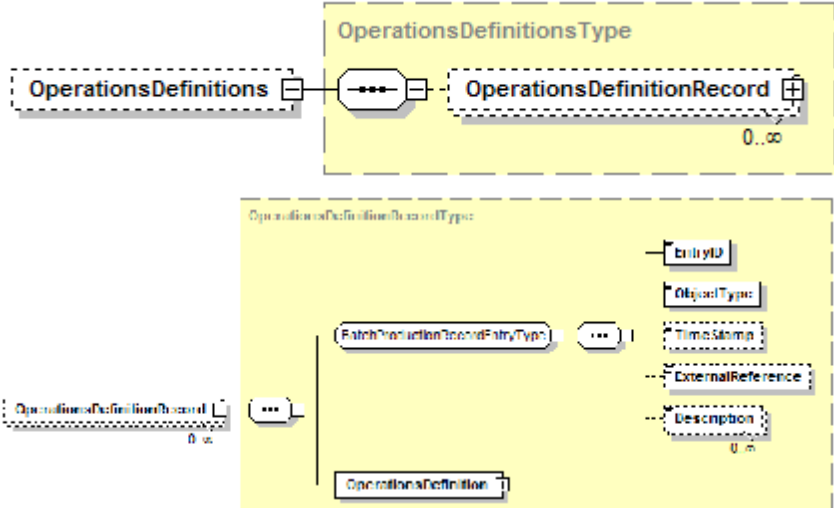
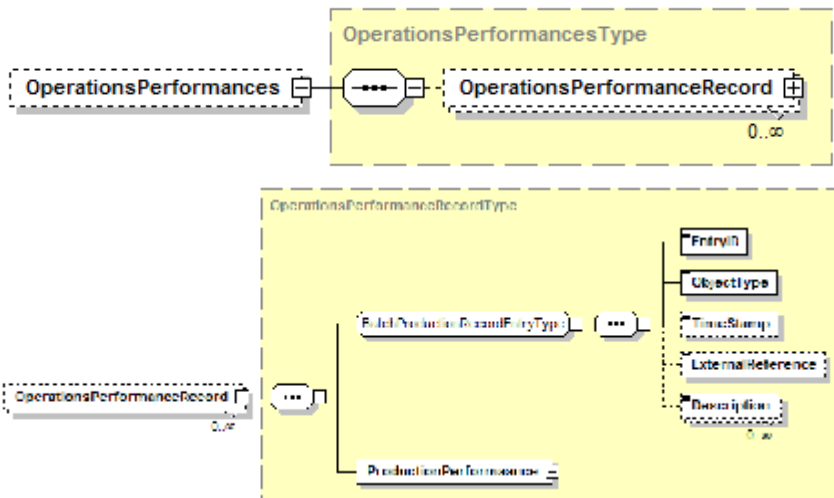


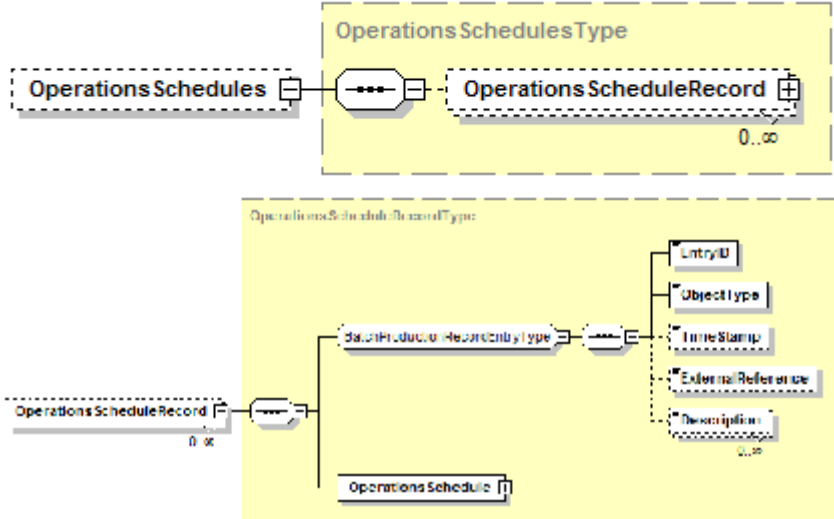
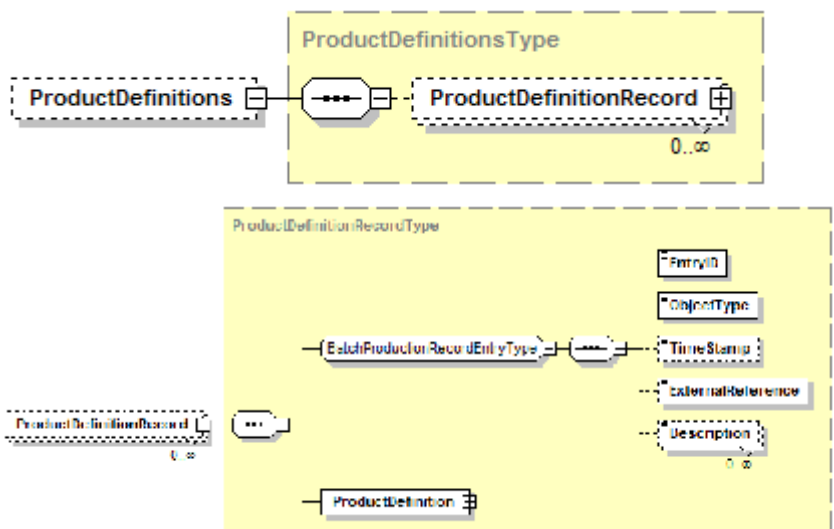
Figure 7 - Batch Production Record Type

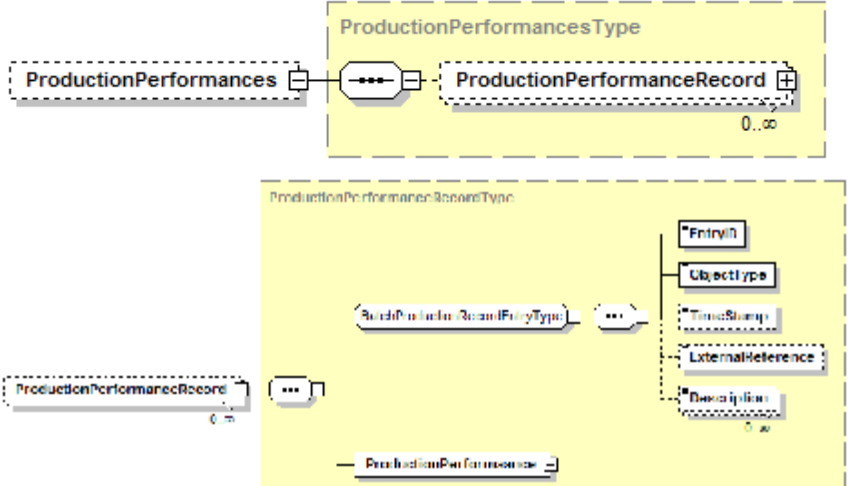
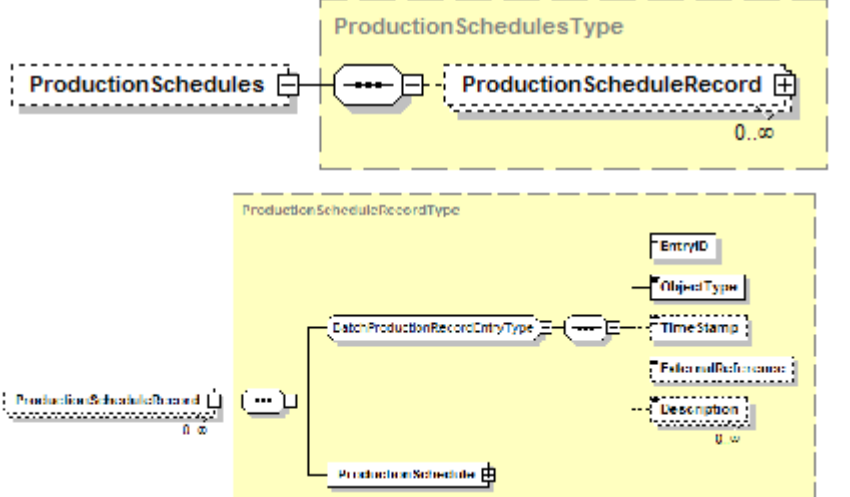
Element/Type	Description
ID IdentifierType	A unique identification of the BPR. This element is mandatory.
ObjectType RecordObjectType	Identifies the type of object an entry is based upon.
TimeStamp DateTimeType	The time stamp associated with the entry.
ExternalReference IdentifierType	Contains a reference to data which is stored external to the BPR.
Description DescriptionType	Additional information about the BPR.
EquipmentScope IdentifierType	The equipment hierarchy scope of the data associated with BPR. This information represents the physical structure of the BPR to identify its context within the plant physical hierarchy
PublishedDate DateTimeType	The date the BPR was published.
CreationDate DateTimeType	The date the BPR was created.
BatchID IdentifierType	The list of IDs of the batches associated with the BPR.
BatchProductionRecordSpec IdentifierType	An optional identification of the Batch Production Record Specification that was used to generate the BPR. The format for this specification is not defined.
CampaignID IdentifierType	An optional identification of the campaign associated with the BPR.
ChangeIndicator xsd:string	An optional indication enabling detection that the batch production record has not been altered. <ul style="list-style-type: none"> Example 1: A string generated by an MD5 algorithm used as a hashing algorithm. Example 2: A string representing a digital key of the entire batch production record. Example 3: A string representing a checksum of the entire batch production record.
Delimiter TextType	Delimiter character used to separate equipment elements in the EquipmentID and PhysicalAssetID elements. Example: "\"
EquipmentID IdentifierType	A definition of the equipment associated with the BPR.
ExpirationDate DateTimeType	The Date and time at which the batch production record is no longer relevant.
Language CodeType	The overall language used in the BPR. Note that and DescriptionTypes or TextTypes have an optional Language attribute. Language codes should be specified using the ISO 639: 1988 specification.
LastChangedDate DateTimeType	The date the BPR was last changed.
LotID IdentifierType	The list of IDs of the lots associated with the BPR.
MaterialDefinitionID IdentifierType	The list of IDs of the Material Definitions associated with the BPR.
PhysicalAssetID IdentifierType	A definition of the physical asset associated with the BPR.
RecordStatus CodeType	Specifies the current status of the BPR, reflecting the current position in the BPRs life cycle. There are no standard codes defined. Example: In Process, In Review, Approved.

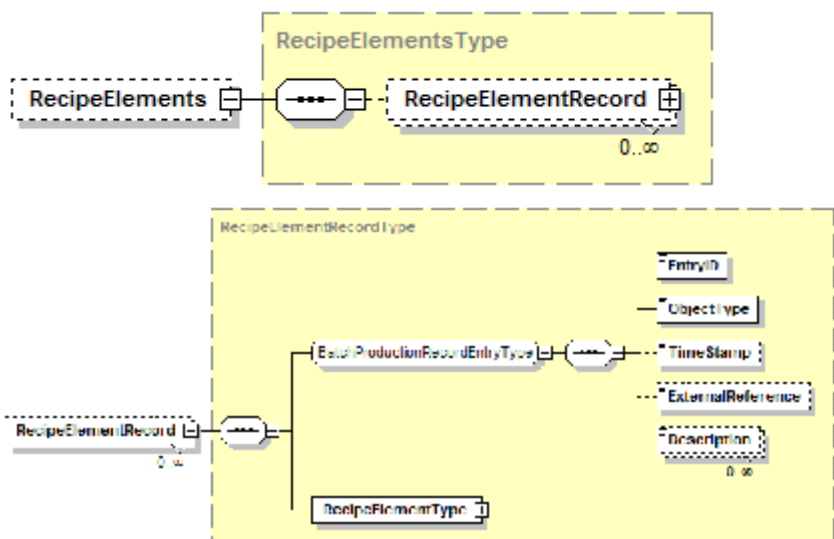
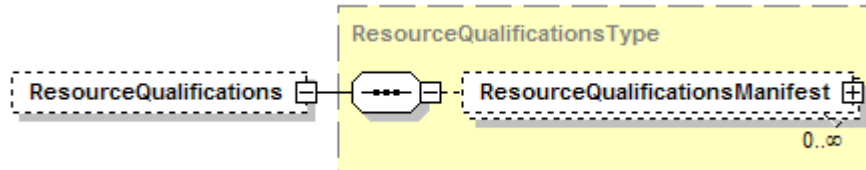
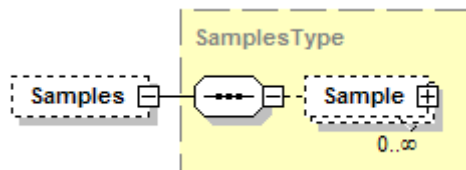
Element/Type	Description
Version IdentifierType	The current version of the BPR.
ChangeHistory ChangeHistoryType	Container type for Change elements. <div data-bbox="597 415 1123 592">  </div>
Comments CommentsType	Container type for Comment elements. <div data-bbox="597 655 1114 831">  </div>
ControlRecipes ControlRecipeTypes	Container type for Control Recipe elements. See BatchML-V0500-BatchInformation for a specification of the ControlRecipe element type. <div data-bbox="597 949 1422 1491">  </div>
DataSets DataSetsType	Container type for DataSet elements. <div data-bbox="597 1545 1058 1722">  </div>

Element/Type	Description
Events EventsTypes	Container type for Event elements. 
MasterRecipes MasterRecipesType	Container type for MasterRecipe elements. See BatchML-V0500-BatchInformation for a specification of the MasterRecipe element type.  
PersonnelIdentification PersonnelIdentificationType	Container type for PersonnelIdentificationManifest elements. 

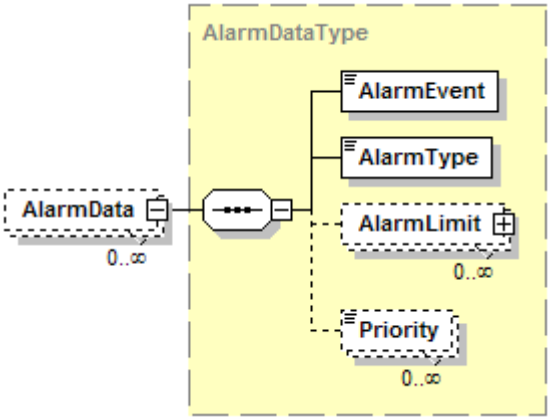
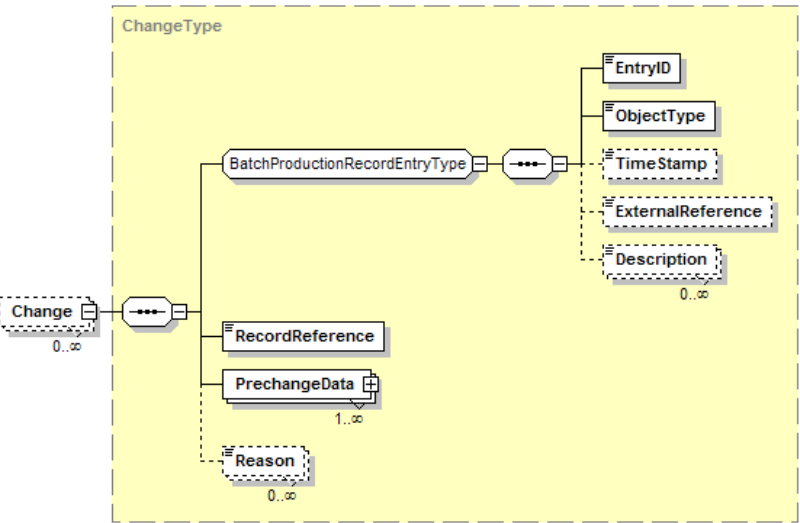
Element/Type	Description
OperationsDefinitions OperationsDefinitionsType	<p>Container type for OperationsDefinition elements. See B2MML-V0500-OperationsDefinition for a specification of the OperationsDefinition element type.</p> 
OperationsPerformances OperationsPerformancesType	<p>Container type for OperationsPerformance elements. See B2MML-V0500-OperationsPerformance for a specification of the OperationsPerformance element type.</p> 

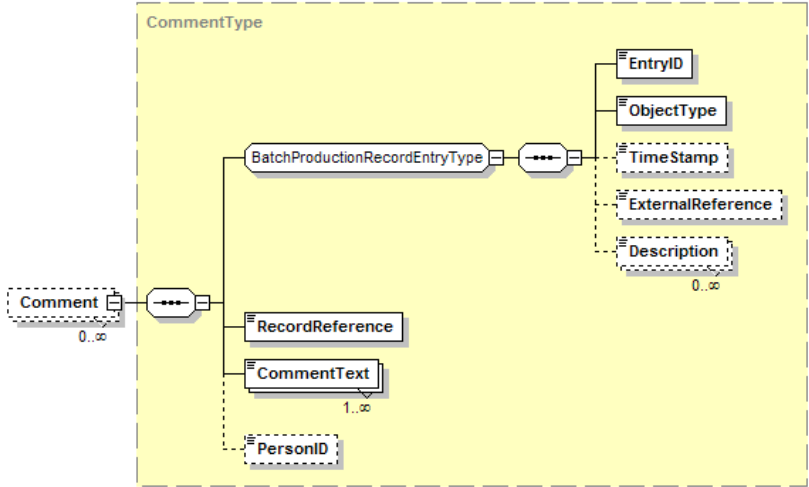
Element/Type	Description
OperationsSchedules OperationsSchedulesType	<p>Container type for OperationsSchedule elements. See B2MML-V0500-OperationsSchedule for a specification of the OperationsSchedule element type.</p> 
ProductDefinitions ProductDefinitionsType	<p>Container type for ProductDefinition elements. See B2MML-V0500-ProductDefinition for a specification of the ProductDefinition element type.</p> 

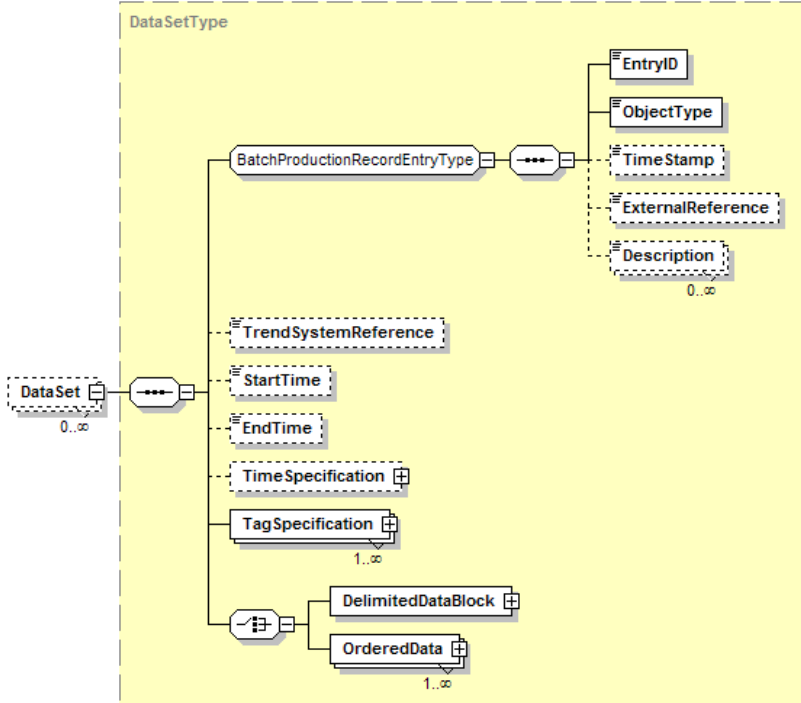
Element/Type	Description
ProductionPerformances ProductionPerformancesType	<p>Container type for ProductionPerformance elements. See B2MML-V0500-ProductionPerformance for a specification of the ProductionPerformance element type.</p> 
ProductionSchedules ProductionSchedulesType	<p>Container type for ProductionSchedule elements. See B2MML-V0500-ProductionSchedule for a specification of the ProductionSchedule element type.</p> 

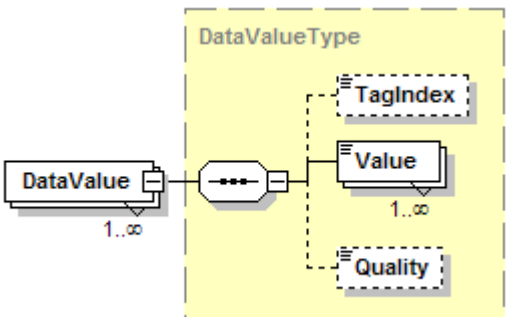
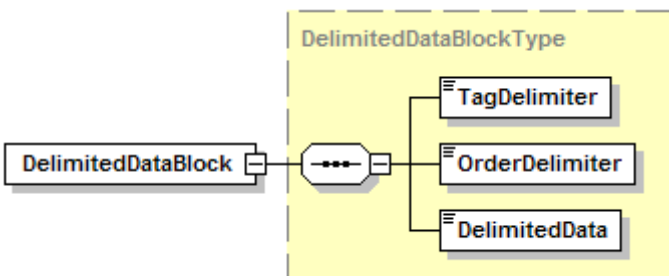
Element/Type	Description
RecipeElements RecipeElementsType	<p>Container type for RecipeElement elements. See BatchML-V0500-BatchInformation for a specification of the RecipeElement element type.</p> 
ResourceQualifications ResourceQualificationsType	
Samples SamplesTypes	
BatchProductionRecord BatchProductionRecordType	Contained BPR.

4.4 Data Elements

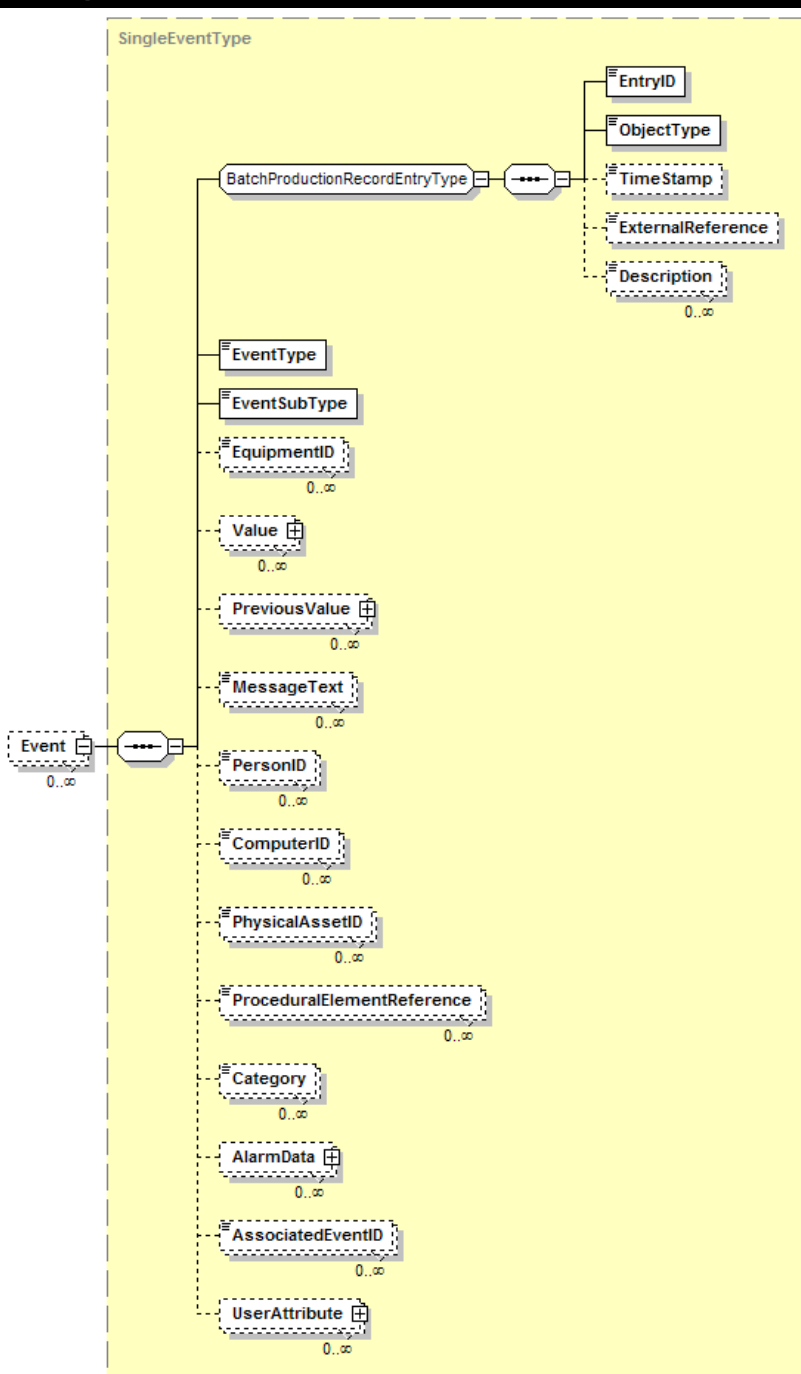
Element/Type	Description
AlarmData AlarmDataType	<p>Defines additional alarm data for an event that is an alarm event.</p> <p>AlarmEvent (CodeType): Defines the type of alarm event. Standard identifiers are not defined. Examples: Detected, Acknowledged, Cleared.</p> <p>AlarmType (CodeType): Defines the type of alarm. Standard identifiers are not defined. Examples: High, Deviation, Rate of Change.</p> <p>AlarmLimit (ValueType): Defines the value measurement that caused the alarm event.</p> <p>Priority (IdentifierType): Defines the importance of the event. Standard identifiers are not defined. Examples: High, Low, 10, 9, 8, 0 ...</p> 
Change ChangeType	<p>A specification of a change to a BPR element. Each Change element contains:</p> <p>RecordReference (IdentifierType): The EntryID of a specific element in the BPR.</p> <p>PreChangeData (ValueType): A definition of the value of a data element prior to change of the data element.</p> <p>Reason (TextType): A text comment with the reason for the change.</p> 

Element/Type	Description
Comment CommentType	<p>A comment related to a BPR element. Each Comment element contains:</p> <p>RecordReference (IdentifierType): The EntryID of a specific element in the BPR.</p> <p>CommentText (TextType): The text comment.</p> <p>PersionID (NameType): Defines the person associated with the comment.</p> 

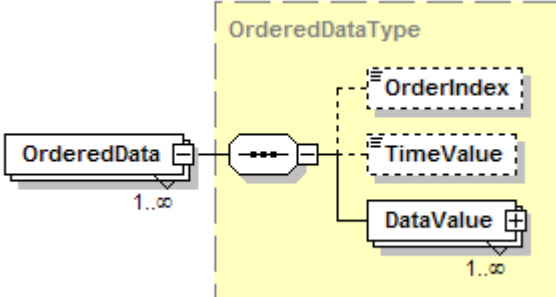
Element/Type	Description
DataSet DataSetType	<p>A data set defines arrays of related data values that can be used to represent time series or correlated data values.</p> <p>Each DataSet element contains:</p> <p>TrendSystemReference (IdentifierType): Specifies the location of the data set if it is stored on an external system.</p> <p>StartTime (DateTimeType): Date and time of the start of data in the data set.</p> <p>EndTime (DateTimeType): Date and time of the end of data in the data set.</p> <p>TimeSpecification (TimeSpecificationType): Defines the attributes of the time specification of the data.</p> <p>TagSpecification (TagSpecificationType):</p> <p>DelimitedDataBlock (DelimitedDataBlockType):</p> <p>OrderedData (OrderedDataTypes):</p> 

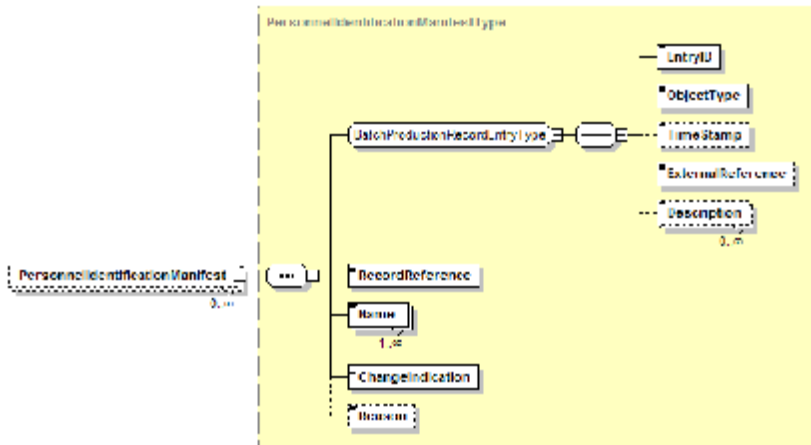
Element/Type	Description
DataValue DataValueType	<p>Defines the value for a data element in a data set.</p> <p>TagIndex (NumericType): Defines a numeric index value the Tag element. NOTE: This is required because the order of repeating elements is not guaranteed in XML.</p> <p>Value (xsd:string): Defines the value of the data for a tag for a data set.</p> <p>Quality (IdentifierType): Defines the quality of the value. This indicates the lack or presence of problems associated with the collection of the data. There are no standard identifiers defined. Example: Good, Uncertain, Out of Date.</p>  <p>The diagram shows a DataValue element connected to a DataValueType container. Inside the container, there are three sub-elements: TagIndex, Value, and Quality. The Value element has a multiplicity of 1..∞.</p>
DelimitedDataBlock DelimitedDataBlockType	<p>Defines a data set in a single delimited string element. The string contains a set of substrings, delimited by the OrderDelimiter, and tag elements in the substring delimited by the TagDelimiter. The selection of the TagDelimiter and OrderDelimiter will be dependent on the data collected. The delimiter characters cannot be part of the data.</p> <p>Note: This data representation is not directly defined in ANSI/ISA 88.04, but is provided to have a more compact form of data representation.</p> <p>TagDelimiter (): Defines the character used to delimit the tag elements within an Order substring. Example: “\”.</p> <p>OrderDelimiter (): Defines the character used to delimit the order substrings. Example “ ”.</p> <p>DelimitedData (): Defines the delimited string with tag elements. Example: “09:10:21\23.4\57\76.2 09:10:25\23.3\57\76.1 09:10:27\23.2\57\76.0”</p> <p>Empty tag data values are indicated by no spaces between TagDelimiters.</p>  <p>The diagram shows a DelimitedDataBlock element connected to a DelimitedDataBlockType container. Inside the container, there are three sub-elements: TagDelimiter, OrderDelimiter, and DelimitedData.</p>
Event SingleEventType	<p>Describes an event. The event contains an EventType, EventSubType, and value. The associated meaning of the event type, subtype, value, and previous value are specified in ANSI/ISA 88.04 and IEC 61512-4.</p> <p>Other elements include:</p> <p>EquipmentID (IdentifierType): Defines the equipment associated with the event.</p> <p>MessageText (TextType): Defines text associated with the event.</p> <p>PersonID (NameType): Defines an identification of the person associated with</p>

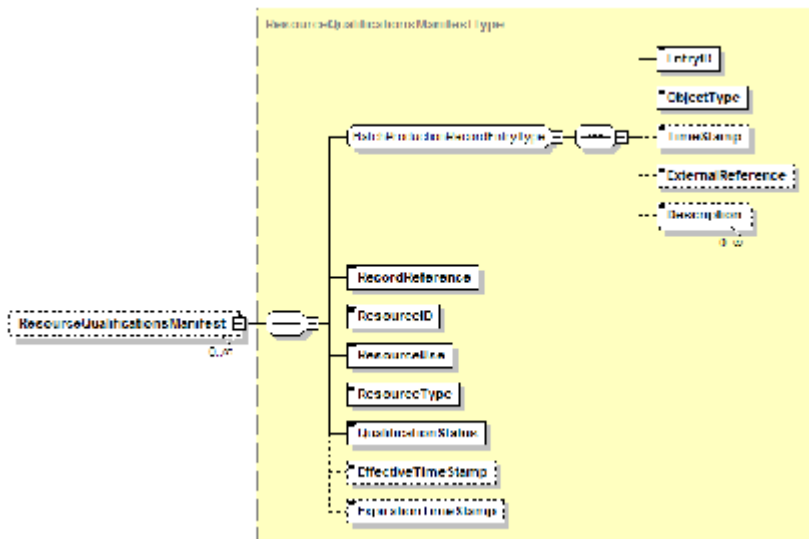
Element/Type	Description
	<p>the event.</p> <p>ComputerID (IdentifierType): Defines the computer or automation system associated with the event.</p> <p>PhysicalAssetID (IdentifierType): Defines the physical asset associated with the event.</p> <p>ProceduralElementReference (IdentifierType): Defines a reference to a procedural element associated with the event, such as a phase or an operation in a control recipe.</p> <p>Category (IdentifierType): Defines the use category of the event. No standard identifiers are defined. Example; Informational, Critical.</p> <p>AlarmData (AlarmDataType): Defines additional alarm data if the event is an alarm type.</p> <p>AssociatedEventID (IdentifierType): Defines the EntryID of any associated event.</p> <p>UserAttribute (UserAttributeType): Defines additional user defined attributes associated with the event.</p>

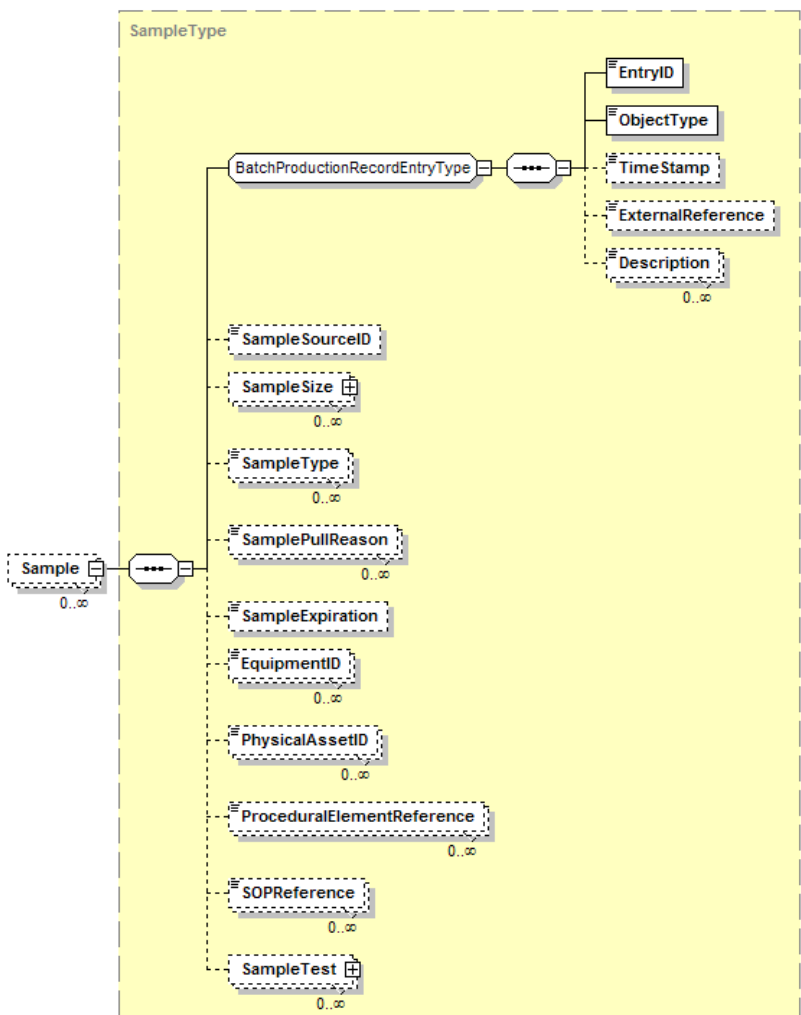
Element/Type	Description
	 <p>The diagram illustrates the structure of the Event element, which is composed of a BatchProductionRecordEntryType and a list of other attributes. The BatchProductionRecordEntryType is further composed of EntryID, ObjectType, TimeStamp, ExternalReference, and Description (0..∞). Other attributes of Event include EventType, EventSubType, EquipmentID (0..∞), Value (0..∞), PreviousValue (0..∞), MessageText (0..∞), PersonID (0..∞), ComputerID (0..∞), PhysicalAssetID (0..∞), ProceduralElementReference (0..∞), Category (0..∞), AlarmData (0..∞), AssociatedEventID (0..∞), and UserAttribute (0..∞).</p>

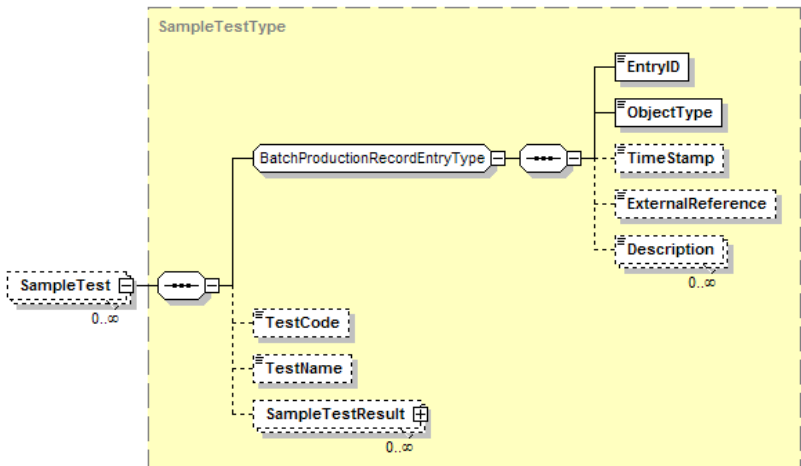
Element/Type	Description
EventSubType EventSubTypeType	<p>Defines an event subtype. The associated meaning of the event type, subtype, value, and previous value are specified in ANSI/ISA 88.04 and IEC 61512-4.</p> <p>This may be either a standard type or an application specific extended type. Standard enumerations correspond to the BPR element types and are:</p> <ul style="list-style-type: none"> • Allocation • Application • Consume • Deallocation • Equipment • Message • Mode Change • Mode Command • Modification • Movement • Parameter Data • Process • Process Data • Produce • Prompt • Prompt Response • Property Value Change • Reconciliation • Security • State Change • State Command • Status Change • System • Target End Time • Target Start Time • Other <p>If "Other" then the type is an application specific extension and the value is defined in the attribute "OtherValue".</p>

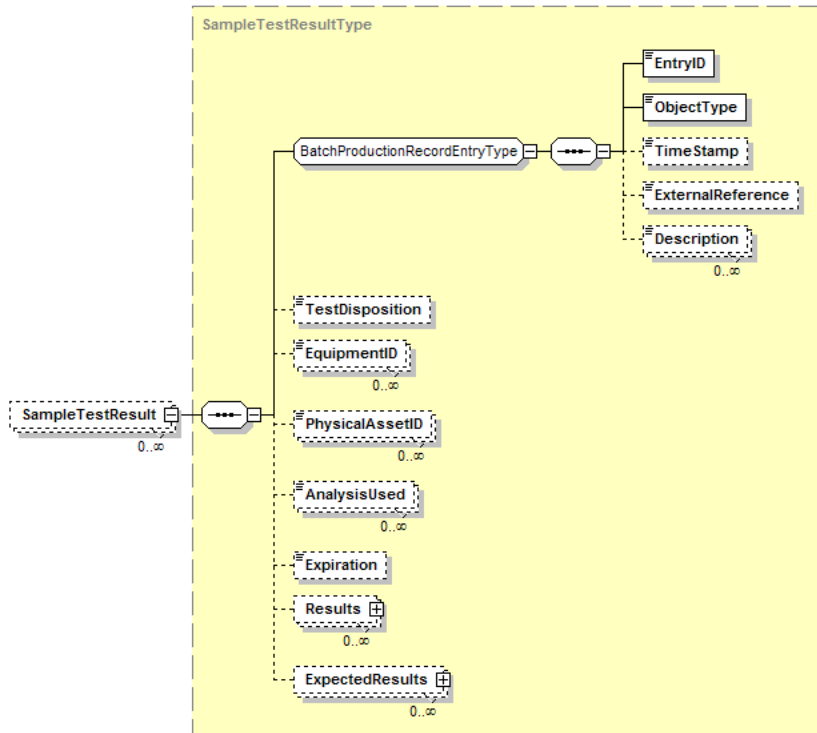
Element/Type	Description
EventType EventTypeType	<p>Defines an event type. The associated meaning of the event type, subtype, value, and previous value are specified in ANSI/ISA 88.04 and IEC 61512-4. This may be either a standard type or an application specific extended type. Standard enumerations correspond to the BPR element types and are:</p> <ul style="list-style-type: none"> • Alarm • Control Recipe • Equipment • General • Material • Message • Operator • Procedural Execution • Other <p>If "Other" then the type is an application specific extension and the value is defined in the attribute "OtherValue".</p>
OrderedData OrderedDataType	<p>Defines a set of ordered data values.</p> <p>OrderIndex (NumericType): Defines a numeric index value the OrderedData element if there is not TimeValue defined. NOTE: This is required because the order of repeating elements is not guaranteed in XML.</p> <p>TimeValue (DateTimeType): Defines the date and time of the OrderedData element if it is time base.</p> <p>DataValue (DataValueType): Defines the value of the data,</p> 

Element/Type	Description
PersonnelIdentificationManifest	<p>Defines the identification of a person associated with a BPR element.</p> <p>RecordReference (IdentifierType): Defines the EntryID of a record in the BPR that the personnel identification manifest is associated with, for example a Change element.</p> <p>Name (NameType): Defines the name of the person.</p> <p>ChangeIndication (xsd:string): An indication enabling detection that the PersonnelIdentificationManifest element has not been altered. Example: string generated by an MD5 hash algorithm of the name, record reference, and reason.</p> <p>Reason (TextType): Defines the reason for the signature.</p>  <pre> classDiagram class PersonnelIdentificationManifestType { EntryID ObjectType TimeStamp ExternalReference Description RecordReference Name ChangeIndication Reason } class PersonnelIdentificationManifest { 0..m } class EntryID class ObjectType class TimeStamp class ExternalReference class Description class RecordReference class Name class ChangeIndication class Reason PersonnelIdentificationManifestType --> EntryID PersonnelIdentificationManifestType --> ObjectType PersonnelIdentificationManifestType --> TimeStamp PersonnelIdentificationManifestType --> ExternalReference PersonnelIdentificationManifestType --> Description PersonnelIdentificationManifestType --> RecordReference PersonnelIdentificationManifestType --> Name PersonnelIdentificationManifestType --> ChangeIndication PersonnelIdentificationManifestType --> Reason PersonnelIdentificationManifestType --> PersonnelIdentificationManifest : 0..m </pre>

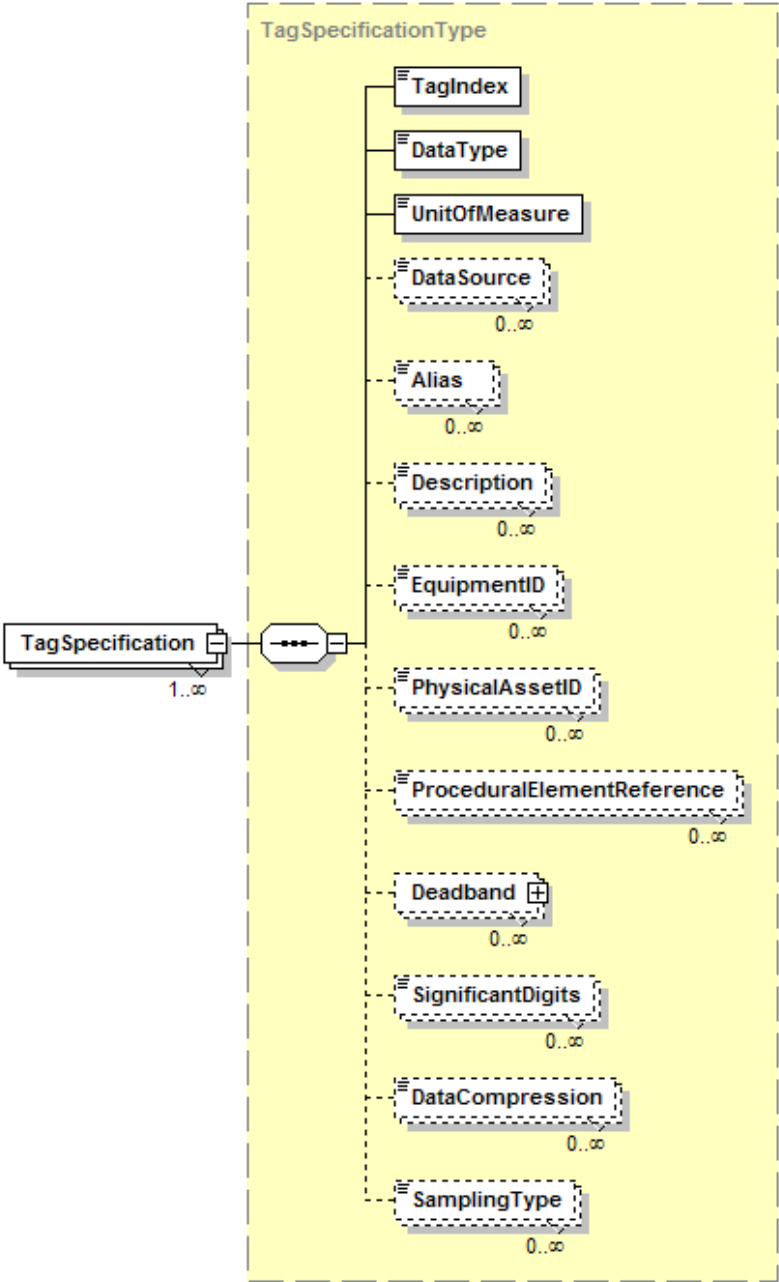
Element/Type	Description
ResourceQualificationManifest	<p>Defines the identification of a resource associated with a BPR element.</p> <p>RecordReference (IdentifierType): Defines the EntryID of a record in the BPR that the resource qualification manifest is associated with</p> <p>ResourceID (IdentifierType): Defines the resource associated with the BPR element.</p> <p>QualificationStatus (CodeType): Defines the criteria of the resource that was recorded. No standard codes are defined. Examples: Sterilized, Cleaned, Services.</p> <p>EffectiveTimeStamp (DateTimeType): Defines the date and time the qualification became effective</p> <p>ExpirationTimeStamp (DateTimeType): Defines the ate and time the qualification expired.</p> 
Sample SampleType	<p>Defines that a material was sampled and/or tested and results of the test</p> <p>SampleSourceID (IdentifierType): Defines the lot, batch, or sample that the sample material was pulled from.</p> <p>SampleSize (QuantityValueType): Defines the amount of material taken for the sample.</p> <p>SampleType (CodeType): Defines the type of sample pulled. There are no standard codes defined. Examples: Receiving Sample, Online Sample, Lab Sample,</p> <p>SamplePullReason (TextType): Defines the reason the sample was pulled.</p> <p>SampleExpiration (DateTimeType): Defines the dat the sample expires.</p> <p>EquipmentID (IdentifierType): Defines a reference to an equipment element that is associated with the sample. Example: The Unit the sample was pulled from.</p> <p>PhysicalAssetID (IdentifierType): Defines a reference to a physical asset element that is associated with the sample.</p> <p>ProceduralElementReference (IdentifierType): Defines a reference to procedural element, such as the phase in a control recipe, that is associated with the sample.</p> <p>SOPReference (IdentifierType): Defines the standard operating procedure (SOP) that is associated with the sample pull.</p>

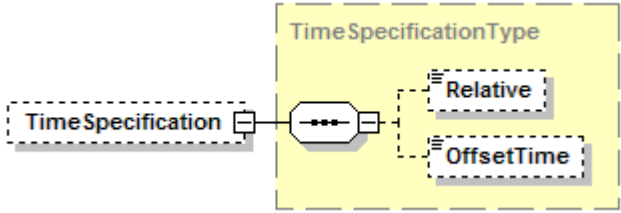
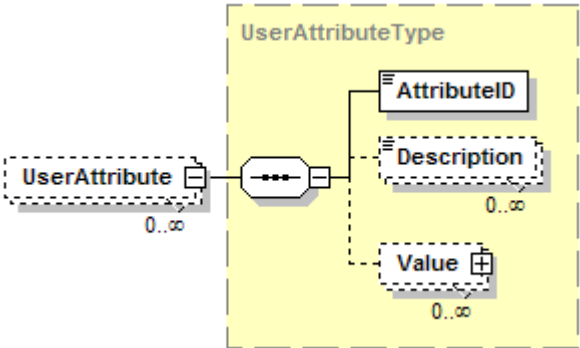
Element/Type	Description
	<p>SampleTest (SampleTestType): Defines the tests on the sample.</p>  <pre> classDiagram class SampleType { BatchProductionRecordEntryType Sample } class BatchProductionRecordEntryType { EntryID ObjectType TimeStamp ExternalReference Description } class Sample { SampleSourceID SampleSize SampleType SamplePullReason SampleExpiration EquipmentID PhysicalAssetID ProceduralElementReference SOPReference SampleTest } SampleType "0..∞" -- "0..∞" BatchProductionRecordEntryType SampleType "0..∞" -- "0..∞" Sample BatchProductionRecordEntryType "0..∞" -- "0..∞" EntryID BatchProductionRecordEntryType "0..∞" -- "0..∞" ObjectType BatchProductionRecordEntryType "0..∞" -- "0..∞" TimeStamp BatchProductionRecordEntryType "0..∞" -- "0..∞" ExternalReference BatchProductionRecordEntryType "0..∞" -- "0..∞" Description Sample "0..∞" -- "0..∞" SampleSourceID Sample "0..∞" -- "0..∞" SampleSize Sample "0..∞" -- "0..∞" SampleType Sample "0..∞" -- "0..∞" SamplePullReason Sample "0..∞" -- "0..∞" SampleExpiration Sample "0..∞" -- "0..∞" EquipmentID Sample "0..∞" -- "0..∞" PhysicalAssetID Sample "0..∞" -- "0..∞" ProceduralElementReference Sample "0..∞" -- "0..∞" SOPReference Sample "0..∞" -- "0..∞" SampleTest </pre>

Element/Type	Description
SampleTest SampleTestType	<p>Defines information on a test of a sample.</p> <p>TestCode (CodeType): defines the test procedure used. Example: CHR3390</p> <p>TestName (identifierType): Defines the name associated with the test performed. Example: Bio Burden</p> <p>SampleTestResult (SampleTestResultType):</p> 

Element/Type	Description
SampleTestResult SampleTestResultType	<p>Defines the result from a sample test.</p> <p>TestDisposition (IdentifierType): Defines an indication if the test results were acceptable. No standard identifiers defined. Example: Pass, Fail</p> <p>EquipmentID (IdentifierType): Defines the equipment used to perform the tests.</p> <p>PhysicalAssetID (IdentifierType): Defines the physical asset used to perform the tests.</p> <p>AnalysisUsed (CodeType): Defines the statistical sampling analysis used to determine the result. No standard CodeTypes defined. Example: Average, Minimum, Maximum</p> <p>Expiration (DateTimeType): Defines the date and time that the sample test results expire.</p> <p>Results (ValueType): Defines the actual value or values returned from the test.</p> <p>ExpectedResults (ValueType): Defines the expected value or values returned from the test.</p> 
TagSpecification TagSpecificationType	<p>Defines a unique identification of a data source. It contains the information to identify a single data source (called a tag) and any data required to interpret the data values, such as data compression information required to use the data.</p> <p>TagIndex (NumericType): An internal number that is used to identify the specific data value in a dataset. (Because the order of repeating elements in XML is not guaranteed, this number is used to tie the data value to the tag specification,)</p> <p>DataType (DataTypeType): From the common elements, defines the data type of the data. (Example: long, unsignedInt, float,...)</p> <p>UnitOfmeasure (UnitOfMeasureType): From the Common elements, defines the unit of measure as a CodeType.</p>

Element/Type	Description
	<p>DataSource (IdentifierType): Defines the data source for the data element, typically the tag name or equivalent.</p> <p>Alias (IdentifierType): Defines an alternate ID of the data source of the data element.</p> <p>Description (DescriptionType): Defines additional information about the data source or the alias identification.</p> <p>EquipmentID (IdentifierType): Defines a reference to equipment, This information represents the physical structure above the referenced entity to identify its context within the plant physical hierarchy.</p> <p>PhysicalAssetID (IdentifierType): Defines a reference to a physical asset.</p> <p>ProceduralElementReference (IdentifierType): Reference to the procedural element, such as procedure, unit procedure, operation, recipe phase, equipment phase, or equipment step associated with the data source ID for the time period of the trend segment.</p> <p>Deadband (ValueType): A deadband value used for collecting and storing data values. If different high and low deadbands are specified, then they should be separate values with keys of HIGH and LOW.</p> <p>SignificantDigits (IdentifierType): Defines the significant digits used for collecting and storing data. This is kept as an identifier type to allow specification or formats for different data types,, such as F2.4 or F7.1 for floating point information and I5 or I8 for integer information.</p> <p>DataCompression (IdentifierType): An identification of the data compression algorithm used when the data was collected. There are no standard identifiers defined. Example; Boxcar Backslope, Change Delta, None.</p> <p>SamplingType (IdentifierType): The type of sample stored. There are no standard identifiers define. Example: Actual/Raw, Interpolated, Best Fit.</p>

Element/Type	Description
	 <p>The diagram illustrates the structure of the TagSpecification element. It is a collection (indicated by the 1..∞ cardinality) of TagSpecificationType objects. The TagSpecificationType is a complex type containing the following attributes:</p> <ul style="list-style-type: none"> TagIndex: A required attribute (solid box). DataType: A required attribute (solid box). UnitOfMeasure: A required attribute (solid box). DataSource: An optional attribute (dashed box) with a cardinality of 0..∞. Alias: An optional attribute (dashed box) with a cardinality of 0..∞. Description: An optional attribute (dashed box) with a cardinality of 0..∞. EquipmentID: An optional attribute (dashed box) with a cardinality of 0..∞. PhysicalAssetID: An optional attribute (dashed box) with a cardinality of 0..∞. ProceduralElementReference: An optional attribute (dashed box) with a cardinality of 0..∞. Deadband: An optional attribute (dashed box) with a cardinality of 0..∞. SignificantDigits: An optional attribute (dashed box) with a cardinality of 0..∞. DataCompression: An optional attribute (dashed box) with a cardinality of 0..∞. SamplingType: An optional attribute (dashed box) with a cardinality of 0..∞.

Element/Type	Description
TimeSpecification TimeSpecificationType	<p>Defines the attributes of the time specification of the data.</p> <p>Relative (IndicatorType): If TRUE then the date/time is relative time, if FALSE then the date/time is absolute time.</p> <p>OffsetTime (DateTimeType): If the time is relative, then the offset defines the absolute time that may be added to the relative times to determine the absolute times.</p>  <p>The diagram shows a dashed box labeled 'TimeSpecification' connected to a central octagonal connector. This connector is linked to a larger dashed box labeled 'TimeSpecificationType'. Inside this box, there are two sub-elements: 'Relative' and 'OffsetTime', each represented by a dashed box with a small icon to its left.</p>
UserAttribute UserAttributeType	<p>Defines user defined attributes and values for events.</p>  <p>The diagram shows a dashed box labeled 'UserAttribute' with a multiplicity of '0..∞' connected to a central octagonal connector. This connector is linked to a larger dashed box labeled 'UserAttributeType'. Inside this box, there are three sub-elements: 'AttributeID' (a solid box), 'Description' (a dashed box with a multiplicity of '0..∞'), and 'Value' (a dashed box with a multiplicity of '0..∞' and a small icon to its left).</p>

5 Diagram Convention

The schema diagrams using the following convention to illustrate the structure of the schema elements, the type of the elements and attributes, and the rules for optional elements and repetition.

