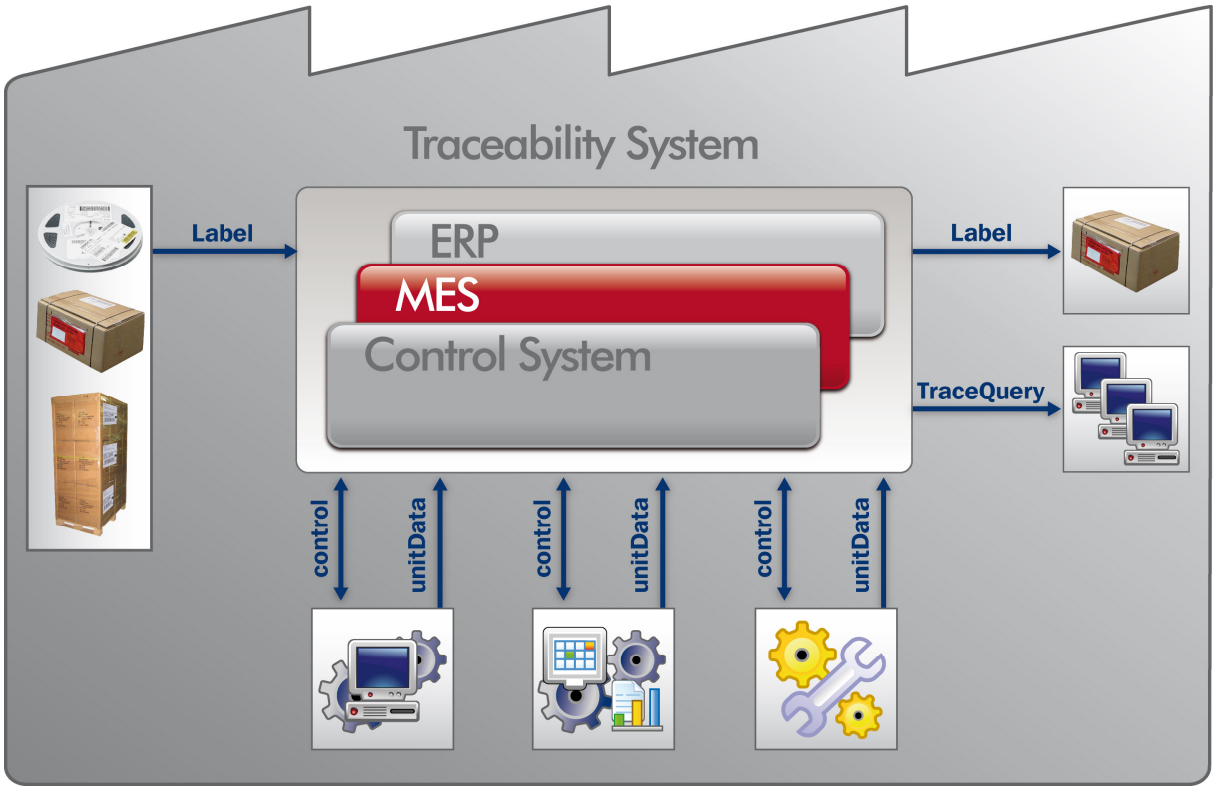
ZVEI
Electronic Components and Systems

Identification and Traceability in the Electrical and Electronics Industry



ZVEI Interfaces to Shopfloor

control

Version 1.1.0

Foreword

Work on the ZVEI manual for the entire supply and value-added chain (see MIT-1 "Guideline for Identification and Traceability) has also served as a draft for an interface to Shopfloor for connecting machines, devices and workstations.

One goal of this undertaking is to standardise the interface for processes in general.

The result was the creation of two XML-based interfaces which are freely available and recommended by ZVEI for connecting to Shopfloor:

- ***control*** for transferring data (requests and return messages) for advanced process control while a product is being processed
- ***unitData*** for transferring processing data of a product

History / changes

In MIT-2 "ZVEI-Interfaces-ChangeHistory" the history of changes of the interfaces ***control*** and ***unitData*** is described.

ZVEI – German Electrical and Electronic
Manufacturers' Association e.V.

Electronic Components and Systems

Lyoner Straße 9

60528 Frankfurt am Main

Fon: 069 6302 – 276

Fax: 069 6302 – 407

Mail: zvei-be@zvei.org

www.zvei-traceability.de

Table of contents

Foreword **2**

History / changes **2**

Table of contents ..... **ii**

1 Introduction **1**

1.1 Symbols used 2

1.2 Legend 2

2 Explanations of XML **3**

2.1 XPath (addressing nodes and attributes) 3

2.2 Serialisers and parsers 4

2.2.1 Formats in the XML file 4

2.2.1.1 Character formats 4

2.2.1.2 Numeric formats 5

2.2.1.2.1 measureDataType 5

2.2.1.3 Date formats 7

3 XML-Schema **8**

3.1 "control-1.1.xsd" 8

3.1.1 <controlRequest> 9

3.1.2 <controlRequestCheck> 9

3.1.3 <controlResponse> 10

3.1.4 [further subnodes] 11

3.1.4.1 [productionResources] 11

3.1.4.2 [processingParameters] 11

3.1.4.3 [properties] 12

3.1.4.4 [setups] 13

3.1.4.5 [assembled] 13

3.1.4.6 [assembledIn] 13

3.1.4.7 [measuringDefinition] 14

3.1.4.8 [testRepairHistory] 14

3.1.4.8.1 [test] 15

3.1.4.8.1.1 subTest/subTestResult 16

3.1.4.8.2 [diagnosis] 17

3.1.4.8.2.1 [subDiagnosis] 18

3.1.4.8.3 [repair] 19

3.1.4.8.3.1 [subRepair] 20

3.1.4.9 [subUnitData] 21

3.1.4.9.1 [further subnodes] ... 21
3.1.4.10 [additionalId] ... 22
3.1.4.11 [additionalData] ... 22
3.1.4.12 [users] ... 22
3.1.4.13 [actions] ... 22

4 XML-Root: control ... 23
4.1 <controlRequest> ... 24
4.2 <controlRequestCheck> ... 26
4.3 <controlResponse> ... 28
4.4 [further subnodes] ... 30
4.4.1 [productionResources] ... 31
4.4.1.1 Resource ... 31
4.4.1.1.1 controlRequest ... 31
4.4.1.1.2 controlRequestCheck, controlResponse ... 31
4.4.2 [processingParameters] ... 32
4.4.2.1 parameter ... 32
4.4.2.1.1 controlRequest ... 32
4.4.2.1.2 controlRequestCheck, controlResponse ... 33
4.4.3 [properties] ... 34
4.4.3.1 <object>Property ... 34
4.4.3.1.1 controlRequest ... 34
4.4.3.1.2 controlRequestCheck, controlResponse ... 35
4.4.4 [setups] ... 36
4.4.4.1 Example for current setups without using setup equipments ... 36
4.4.4.1.1 controlRequest ... 36
4.4.4.1.2 controlRequestCheck, controlResponse ... 37
4.4.4.2 Example for nominal setup data using setup equipments ... 38
4.4.4.2.1 controlRequest ... 38
4.4.4.2.2 controlRequestCheck, controlResponse ... 38
4.4.4.2.3 [productionReferences] ... 39
4.4.4.2.4 [setupEquipment] ... 39
4.4.4.2.4.1 [material] ... 39
4.4.5 [assembled] ... 40
4.4.5.1 controlRequest ... 40
4.4.5.2 controlRequestCheck ... 40
4.4.5.3 controlResponse ... 40
4.4.6 [assembledIn] ... 41
4.4.6.1 controlRequest ... 41
4.4.6.2 controlRequestCheck ... 41

4.4.6.3 controlResponse ... 41
4.4.7 [measuringDefinition] ... 42
4.4.7.1 channelDefinition ... 42
4.4.7.1.1 controlRequest ... 42
4.4.7.1.2 controlRequestCheck, controlResponse ... 42
4.4.8 [testRepairHistory] ... 43
4.4.8.1 controlRequest ... 43
4.4.8.2 controlRequestCheck ... 43
4.4.8.3 controlResponse ... 44
4.4.8.3.1 [test] ... 45
4.4.8.3.2 [diagnosis] ... 46
4.4.8.3.3 [repair] ... 47
4.4.9 [subUnitData] ... 48
4.4.9.1 controlRequest ... 48
4.4.9.2 controlRequestCheck, controlResponse ... 49
4.4.9.3 [further subnodes] ... 49
4.4.10 [additionalId] ... 50
4.4.10.1 controlRequest, controlRequestCheck, controlResponse ... 50
4.4.11 [additionalData] ... 51
4.4.11.1 controlRequest, controlRequestCheck, controlResponse ... 51
4.4.11.1.1 data ... 51
4.4.12 [users] ... 52
4.4.12.1 controlRequest ... 52
4.4.12.2 controlRequestCheck, controlResponse ... 52
4.4.12.2.1 user ... 52
4.4.13 [actions] ... 53
4.4.13.1 controlRequest, controlRequestCheck, controlResponse ... 53
4.4.13.1.1 action ... 53
4.4.13.1.1.1 [expression] ... 53
4.4.13.2 Examples ... 54

5 Appendix ... 55
5.1 List of relevant documents ... 55
5.2 List of Figs ... 56
5.3 List of tables ... 58
5.4 List of relevant terms and abbreviations ... 59

1 Introduction

This document specifies the structure of the standardZVEI interface ***control***. This interface describes the transfer of data (requests and responses) for advanced process control during the processing of a product.

The interface may be utilised for process interlocking or for the querying of data. The requests for process control are handled by the subnodes 'controlRequest' or 'controlRequestCheck'. The subnode 'controlResponse' returns the results or forwards information to other processes.

Various transfer protocols may be used for transferring the XML structures. A detailed specification of these protocols can be found in MIT-7 "ZVEI interface transfer protocols".

The XML files described here (see MIT-12 "control_Request-1.1.xml" and MIT-13 "control_Response-1.1.xml") serve as examples for the use of an XSD schema file (seeMIT-8 "control-1.1.xsd"). For any specific use of the interface, the contents of the XML structures will have to be agreed between the parties who are to carry out the communication.

The structures consist of:

A uniform cover sheet controlRequest, controlRequestCheck oder controlResponse with the attributes

- requestID respectively responseToRequestID ➔ unique Telegram-IDs
- optional attributes ➔ e.g. unit, order, material
- Status of request respectively response ➔ state

Optional attachment sheets (similar to interface unitData) for

- Tools and production resources ➔ productionResources
- Parameters and nominal values ➔ processingParameters
- Additional properties ➔ properties
- Nominal and actual setups ➔ setups
  Materials that should be consumed
  in the next process step
- Set points and limiting values for measurements ➔ measuringDefinition
- Test, diagnostic and repair data ➔ testRepairHistory (test, diagnosis, repair)
- Handling of workpiece holders ➔ subUnitData
- Authentication of operators ➔ users
- Additional data ➔ additionalData
- Actions to be performed by recipient ➔ actions

1.1 Symbols used

Three different symbols are used in this documentation to emphasise important content items.

Attention!

This symbol refers to important information for which compliance is absolutely mandatory.

Explanation!

This symbol refers to explanatory information.

Tip!

This symbol identifies tips which provide faster or more efficient solutions.

1.2 Legend

[Node/attribute **]** Square brackets: ➔ optional node/attribute

< Node/attribute **>** Pointed brackets: ➔ alternative node/attribute

Explanation!

If an attribute is required (not optional), the value must be assigned (no empty string).

If an attribute is not required (optional) but is present with the value = "" (empty string), the attribute will be ignored. The attribute will then be handled as if it were not even present.

2 Explanations of XML

Data is transferred via standard compliant XML structures. The layout and format of XML structures for each interface are saved in an XSD schema file.

Explanation!

All information about XML can be found in http://www.w3.org/XML/. XML specifications with translations into various languages are also available at the same location.

Additional sites with information related to this topic are:

- WIKIPEDIA, the free encyclopaedia http://de.wikipedia.org/wiki/XML

2.1 XPath (addressing nodes and attributes)

The XML Path Language (XPath) is a query language developed by the W3 consortium to address parts of an XML document. An XPath expression addresses parts of an XML document, which is considered as a tree.

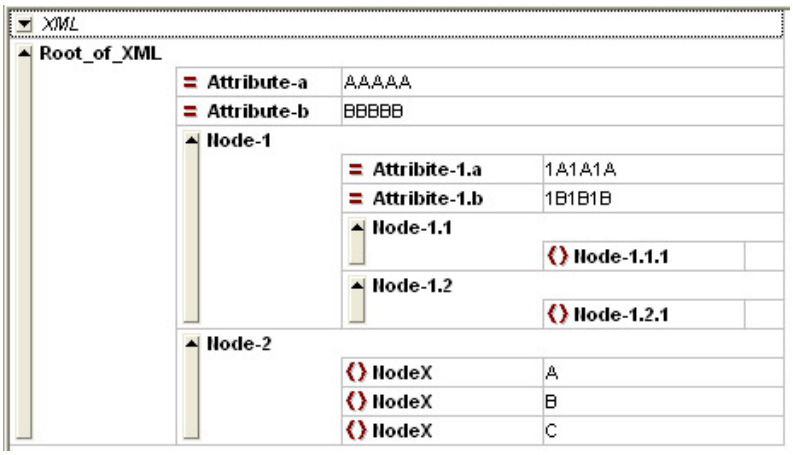


Fig 1: Example of an XPath

Examples of XPath expressions for the XML structure from Fig 1:

- **/Root_of_XML** selects the root element "Root_of_XML" of the XML structure.
- **/*** selects the root element independently of the name
  (every well formed XML document has exactly one root element)
- **/Root_of_XML /Node-2/NodeX** selects *all* "NodeX" elements within the "Node-2" node.
  In this example 3 elements are addressed: NodeX=A, NodeX=B, NodeX=C
- **child::*** selects all child elements of the current node
- **child:: NodeX** selects all "NodeX" children of the current node
- **./*** selects all subelements of the current node
- **/Root_of_XML/attribute::Attribute-a** (abbreviated notation **/Root_of_XML/@Attribute-a**) selects the attribute "Attribute-a" of root element "Root_of_XML" and addresses the value "AAAA"
- **/Root_of_XML/attribute::*** (abbreviated notation **/Root_of_XML/@***) selects all attributes of root element "Root_of_XML"
- **attribute::Attribute-1.a** (abbreviated notation **@Attribute-1.a**) selects all attributes "Attribute-1.a" of the current node
- **attribute::*** (abbreviated notation **@***) selects all attributes of the current node

Explanation!

All information related to XPath can be found at http://www.w3.org/TR/xpath20.

Additional sites with information related to this topic are:

- ZVON.org, The Guide to the XML Galaxy: http://www.zvon.org/xxl/XPathTutorial
- WIKIPEDIA, the free encyclopaedia: http://de.wikipedia.org/wiki/XPath

2.2 Serialisers and parsers

When sending, receiving and processing XML structures, current format-supporting XML processing tools must be used. XML tools must also be taken into consideration in the XSD schema file specified in the XML file. This ensures compliance with XML specifications.

- XML serialisers check compliance with XML specifications when data structures are sent
- XML parsers check compliance with XML specifications when data structures are received

Attention!

Before an interface is integrated, the sender must check to ensure compliance with XML specifications using a current XML parser, XML editor or XML checker. The XML tool that is used must also check to ensure the XSD schema file complies with the XML schema.

2.2.1 Formats in the XML file

The format of data transferred as an XML structure must comply with the requirements of the XML schema in the XSD schema file.

2.2.1.1 Character formats

The character set for character formats is optionally defined as the "encoding" attribute in the first node of the XML structure.

Fig 2: XML encoding

Examples of the "encoding" attribute:

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml version="1.0" encoding="iso-8859-1"?>
```

Explanation!

The standard character set is loaded without the "encoding" attribute being specified.

Attention!

If the character sequences are intended to include special characters and umlauts, it is essential to specify the corresponding character set in the "encoding" attribute.

2.2.1.2 Numeric formats

Attention!

In general the number formats are given from the XSD schema files

In addition to the XML standard formats long and double in the XSD schema file ZVEI Version.xsd-common-specific formats are defined as simpleType.

Available number formats:

Format	Source	Description
long	Standard XML	Natural number together with negatives Examples: … -4; -3; -2; -1; 0; 1; 2; 3; …
double	Standard XML	Floating point number Examples: -1.5; -1; -0.5; 0; 0.5; 1; 1.5; 3.1415; 2.71828 **Explanation**: In the double floating-point format a point is to be used as the decimal point (international notation) and not a comma as in the German spelling.
positiveDouble	ZVEI-common simpleType	Floating point number > 0 Beispiele: 0.5; 1; 1.5; 3.1415; 2.71828 **Explanation:** The number 0 is not permitted.
measureDataType	ZVEI-common simpleType	Value in a special notation (see 2.2.1.2.1 measureDataType)

2.2.1.2.1 measureDataType

In some cases it makes sense for the sake of resolution or readability to display a value in a certain notation.

This can be done with the simpleType measureDataType. The attribute "measureDataType" holds the type of data and the attribue "value" holds the value. Example: measureDataType="decimal" value="3.1415".

Measurement and numerical values with the attribute "measureDataType" can have the following data types:

decimal, exponential, metricPrefix, hexadecimal, binary, string

Available measureDataType

measureDataType	Example	Description
decimal	0.031	Integer or floating point number **Explanation:** In the decimal format a point is to be used as the decimal point (international notation) and not a comma as in the German spelling.
exponential	3.1E-2	Integer or floating point number, and directly afterwards the symbol E for exponent and the exponent itself as an integer **Explanation:** A negative exponent as a minus sign. A positive exponent as a plus sign or not sign. **Explanation** Between the number, and the symbol E, and the exponent no caracters (also no white spaces!) are allowed
metricPrefix	1µ (oder auch 31u)	Integer or floating point number, and directly afterwards the metric prefix. **Explanation** Between the number, and the metric prefix no characters (also no white spaces!) are allowed

Prefix	Symbol	Multiplier	Exp
yotta	Y	1,000,000,000,000,000,000,000,000	10^{24}
zetta	Z	1,000,000,000,000,000,000,000	10^{21}
exa	E	1,000,000,000,000,000,000	10^{18}
peta	P	1,000,000,000,000,000	10^{15}
tera	T	1,000,000,000,000	10^{12}
giga	G	1,000,000,000	10^{9}
mega	M	1,000,000	10^{6}
kilo	k	1	10^{3}
hecto	h	100	10^{2}
deca	da	10	10^{1}
		1	10^{0}
deci	d	0.1	10^{-1}
centi	c	0.01	10^{-2}
milli	m	0.001	1^{0-3}
micro	µ (u)	0.000001	10^{-6}
nano	n	0.000000001	10^{-9}
pico	p	0.000000000001	10^{-12}
femto	f	0.000000000000001	10^{-15}
atto	a	0.000000000000000001	10^{-18}
zepto	z	0.000000000000000000001	10^{-21}
yocto	y	0.000000000000000000000001	10^{-24}

hexadecimal	1F	Integer in hexadecimal notation
binary	00011111	Integer in binary notation
string	any string	Arbitrary sequence of characters

2.2.1.3 Date formats

Dates and times must be indicated in the ISO 8601-compliant format. The optional delimiters according to the ISO standard (-[hyphen] when writing a date, :[colon] when writing a time) must be fully indicated. Dates must be written using the following notation:

YYYY-MM-DD

Times must always be specified in "Coordinated Universal Time" (UTC) using the following notation:

hh:mm:ss

Attention!

The value range for representing seconds is 00 - 60! This makes it possible to represent leap seconds.

If a time is requested (for example "starttime"), it must be fully indicated, including the time zone information, as follows:

YYYY-MM-DDThh:mm:ss+hh:mm

Examples:

2009-12-01T11:01:00+01:00
means December 1, 2009, 11:01:00 in German local time (standard time).
This corresponds to December 1, 2009 10:01:00 in UTC.
+01:00 means + 1 hour compared to coordinated universal time (UTC).

2009-07-03T11:01:00+02:00
means July 3, 2009, 11:01:00 in German local time (daylight saving time).
This corresponds to July 3, 2009 09:01:00 in UTC.
+02:00 means + 2 hours compared to coordinated universal time (UTC).

A description of the ISO 8601-compliant time stamp format is available from:

http://www.cl.cam.ac.uk/~mgk25/iso-time.html

http://www.w3.org/TR/xmlschema-2/#isoformats
http://www.w3.org/TR/NOTE-datetime

Attention!

Time synchronisation is required between the communication partners. A suitable time server must be made available by the system operator for this purpose.

3 XML-Schema

The XML schema is stored in an XSD file.

3.1 "control-1.1.xsd"

The illustration below shows the schema of the ZVEI interface ***control*** (see MIT-8 "control-1.1.xsd").

The interface may be used for process release/process interlocking or querying of data. The subnodes 'controlRequest' or 'controlRequestCheck' are used by requests for process control. The subnode 'controlResponse' is used for the response of results or the forwarding of information to other processes.

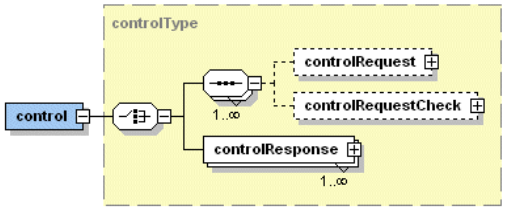


Fig 3: "control-1.1.xsd": Schema

Explanation!

The XSD schema file references another XSD schema file called "ZVEI-common-1.1.xsd" (see MIT-10 "ZVEI-common-1.1.xsd"). Both of these files must be located as references in the same directory

Explanation!

The XSD schema file "ZVEI-testRepair-1.1.xsd" for forwarding test and repair data has been incorporated in the XSD schema file (see MIT-11 "ZVEI-testRepair-1.1.xsd"). Both XSD schema files have to be located in the same directory for reference purposes.

Explanation!

The "controlRequest" node is a request for preset values and parameters. The "controlRequestCheck" node is a request for a check on values and parameters.

Attention!

The "controlRequest" and "controlRequestCheck" nodes may be present several times in the XML structure (XML file). The answers to these requests are contained in the "controlResponse" node of the XML response structure (XML response file). The "requestID" attributes and its sister attributes "responseToRequestID" serve as a link between the various requests and responses.

3.1.1 <controlRequest>

The requests for data, which is to be used during the processing of a product, are handled by the 'controlRequest' subnode.

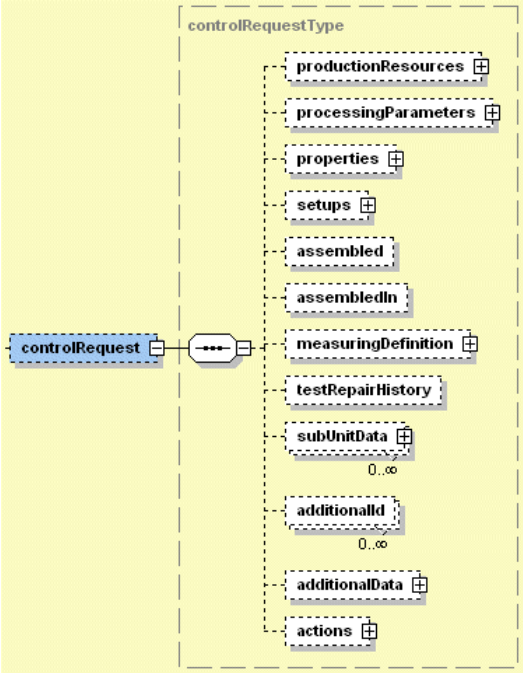


Fig 4: "control-1.1.xsd": Schema controlRequest

3.1.2 <controlRequestCheck>

Requests for process interlocking are handled by the 'controlRequestCheck' subnode.

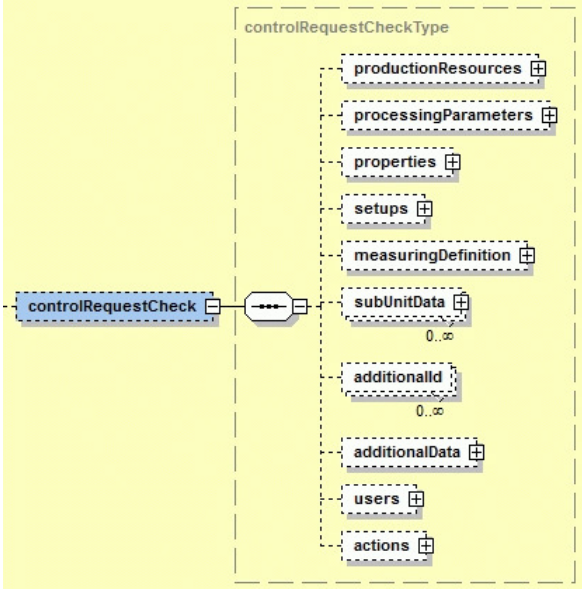


Fig 5: "control-1.1.xsd": Schema controlRequestCheck

3.1.3 <controlResponse>

The subnode "controlResponse" is used to respond to requests, which were either sent using the “controlRequest” or the “controlRequestCheck” node.

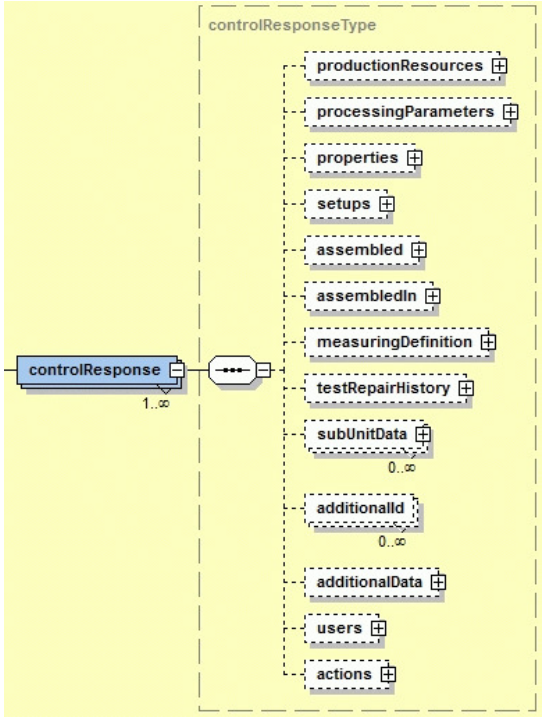


Fig 6: "control-1.1.xsd": Schema controlResponse

3.1.4 [further subnodes]

The alternative nodes "controlRequest", "controlRequestCheck" and "controlResponse" may optionally contain further subnodes. The schema for these subnodes differ in terms of attributes, depending on whether they are used within "controlRequest", "controlRequestCheck" or "controlResponse" node. The following examples of these subnodes apply to the subnode "controlResponse".

Optional attachment sheets (similar to interface unitData) for

- Tools and production resources ➔ productionResources
- Parameters and nominal values ➔ processingParameters
- Additional properties ➔ properties
- Nominal and actual setups ➔ setups
  Materials that should be consumed
  in the next process step
- Set points and limiting values for measurements ➔ measuringDefinition
- Test, diagnostic and repair data ➔ testRepairHistory (test, diagnosis, repair)
- Handling of workpiece holders ➔ subUnitData
- Authentication of operators ➔ users
- Additional data ➔ additionalData
- Actions to be performed by recipient ➔ actions

3.1.4.1 [productionResources]

Tools and production resources that are to be used in the upcoming process step.

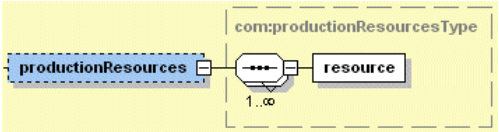


Fig 7: "control-1.1.xsd": Schema node productionResources

3.1.4.2 [processingParameters]

Parameters and nominal values that are to be used in the upcoming process step.

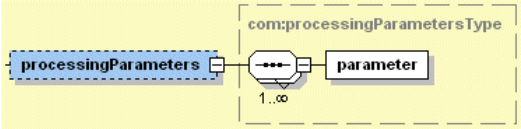


Fig 8: "control-1.1.xsd": Schema node processingParameters

3.1.4.3 [properties]

Additional properties for processing a product.

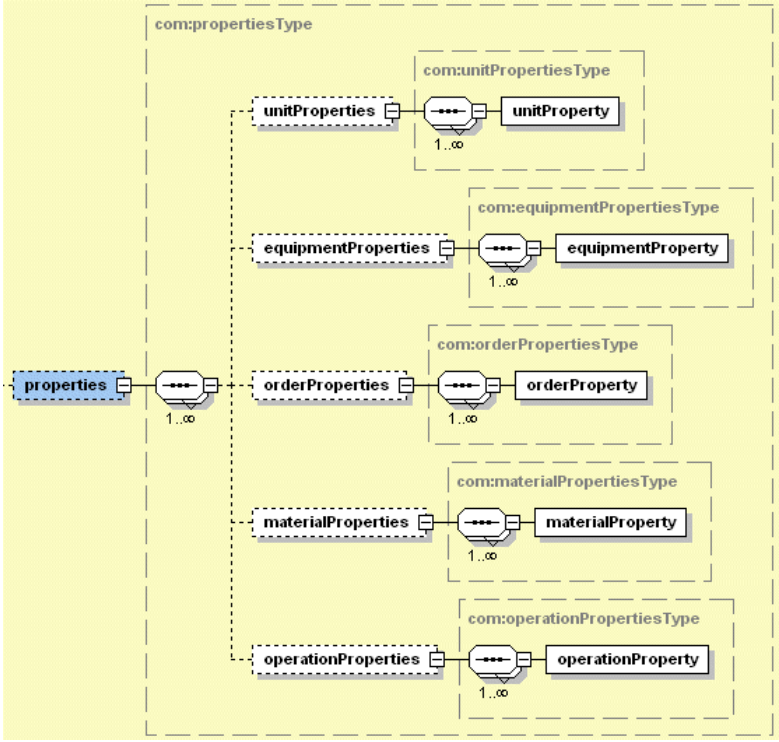


Fig 9: "control-1.1.xsd": Schema node properties

Additional properties

- for the unit (serial number of the material being processed)
- for the equipment (machine, manual workstation, production line or production cell)
- for the order
- for the material (product)
- for the operation

3.1.4.4 [setups]

Setup data or materials, which are to be assembled in the upcoming process step.

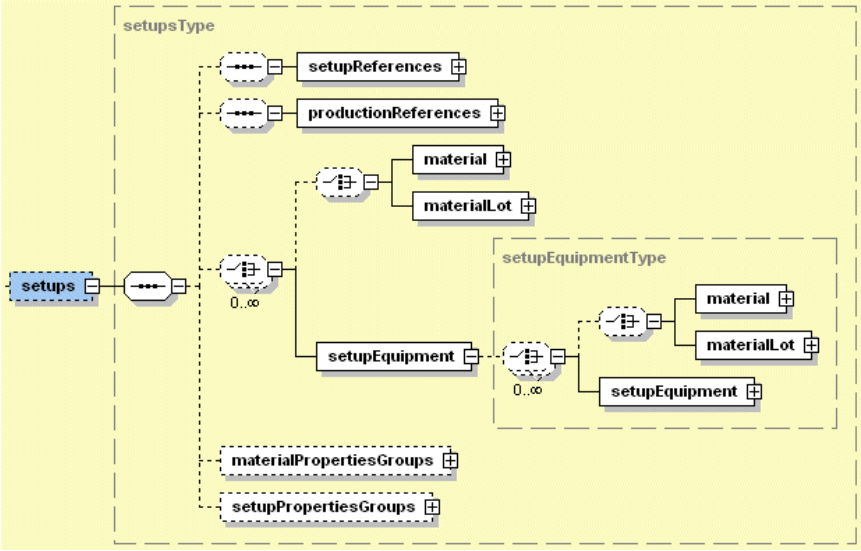


Fig 10: "control-1.1.xsd": Schema node setups

3.1.4.5 [assembled]

Data about materials, which have been assembled in this unit.

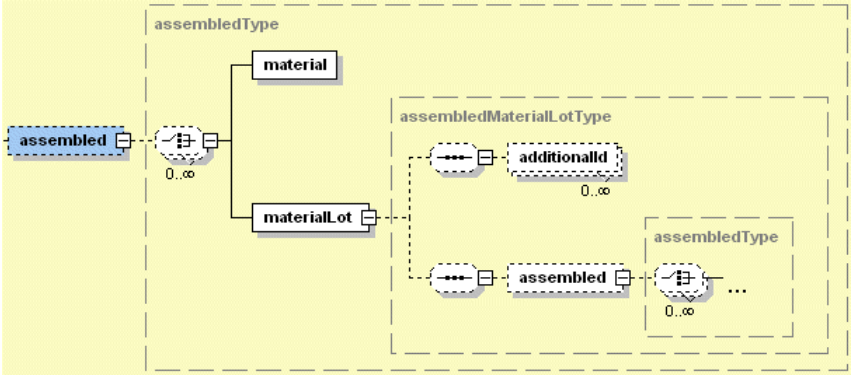


Fig 11: "control-1.1.xsd": Schema node assembled

3.1.4.6 [assembledIn]

List of parent units where the processed material is installed.

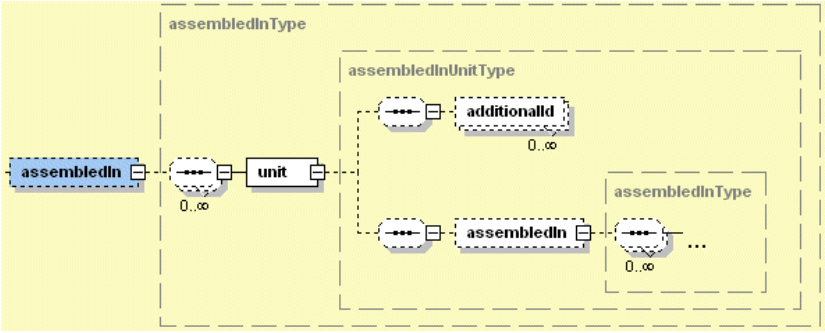


Fig 12: "control-1.1.xsd": Schema node assembledIn

3.1.4.7 [measuringDefinition]

Nominal and boundary values for measurements.

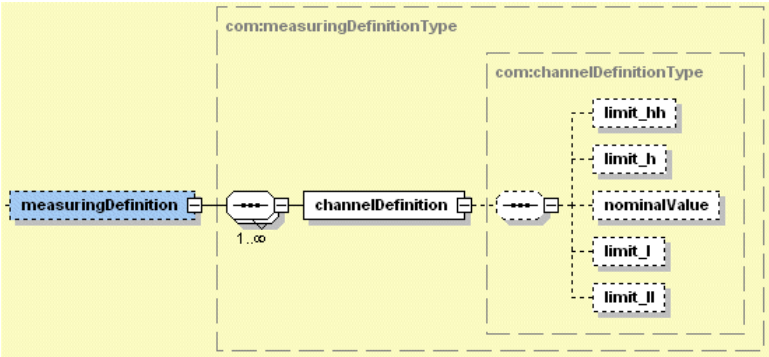


Fig 13: "control-1.1.xsd": Schema node measuringDefinition

3.1.4.8 [testRepairHistory]

The history of test and repair data.

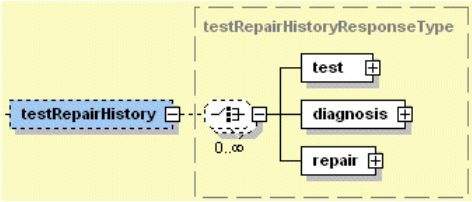


Fig 14: "control-1.1.xsd": Schema node testRepairHistory

3.1.4.8.1 [test]

The history of test data.

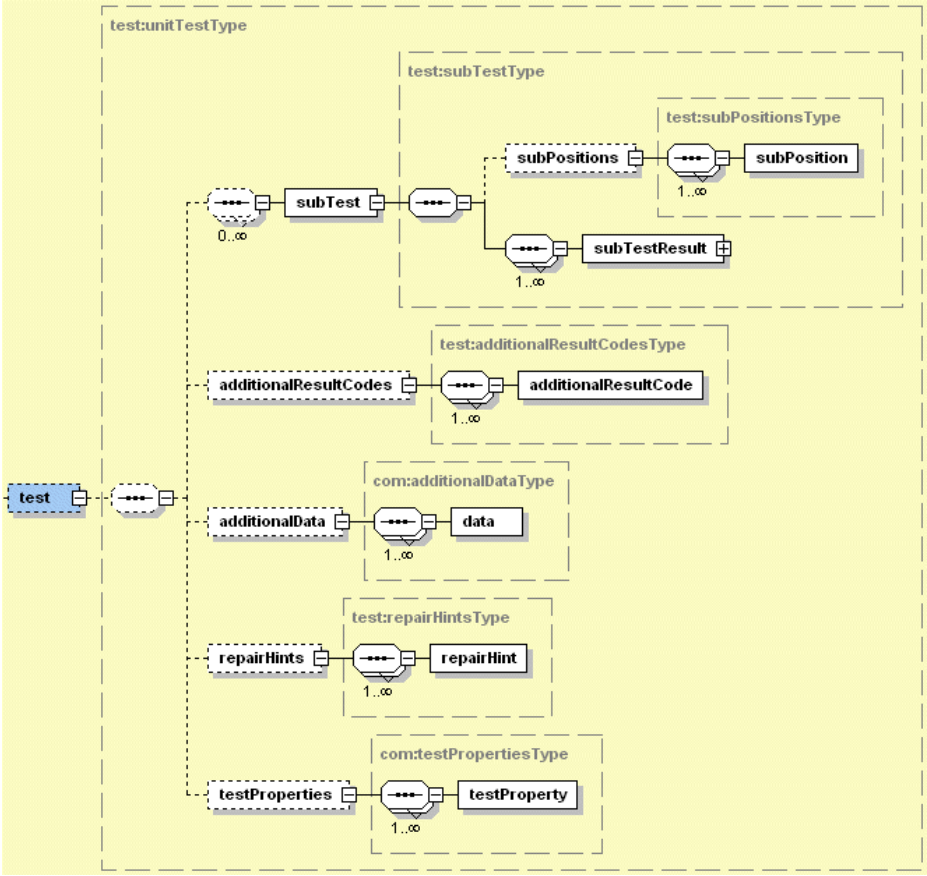


Fig 15: "control-1.1.xsd": Schema node testRepairHistory/test

3.1.4.8.1.1 subTest/subTestResult

Details about the results of a test, which has been performed.

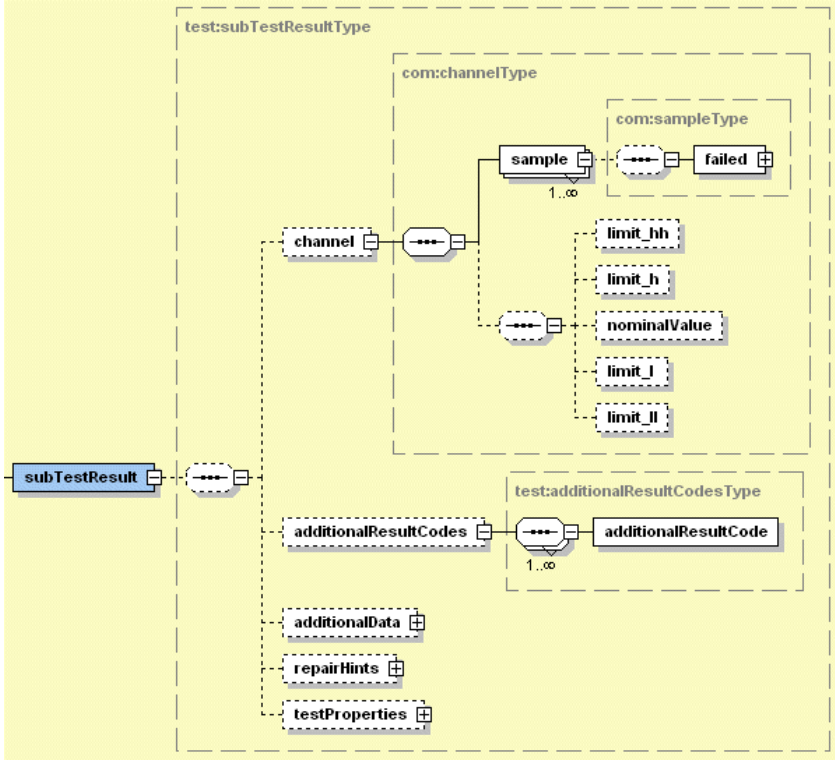


Fig 16: "control-1.1.xsd": Schema test/subTest/subTestResult

3.1.4.8.2 [diagnosis]

History of diagnostic data (analysis, classification) for test results.

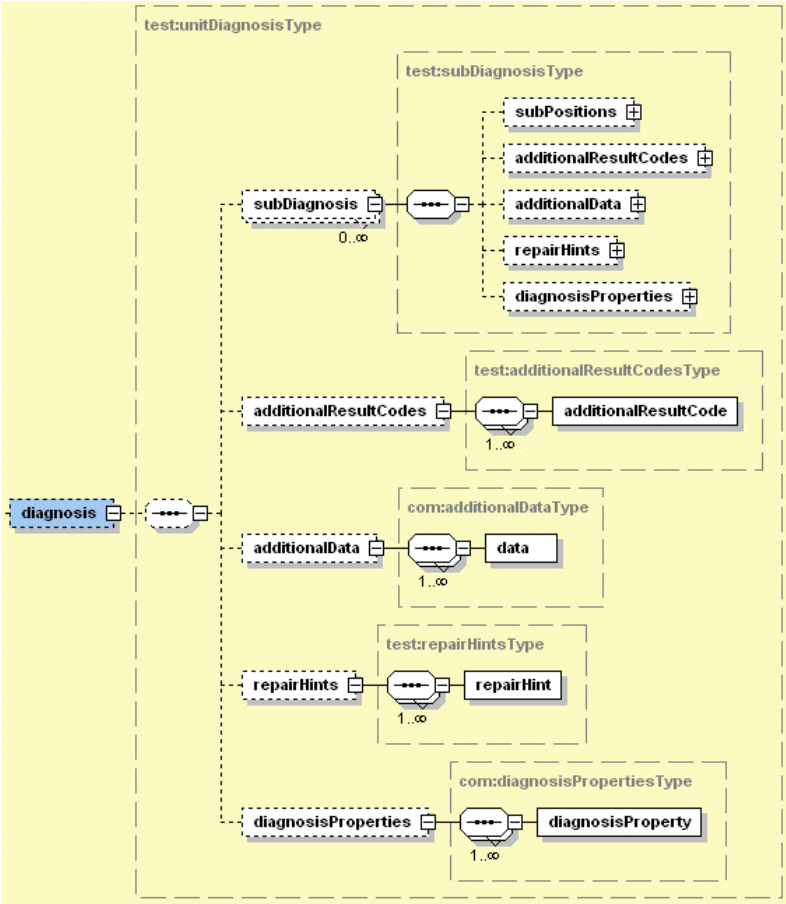


Fig 17: "control-1.1.xsd": Schema node testRepairHistory/diagnosis

3.1.4.8.2.1 [subDiagnosis]

Diagnostic data (analysis, classification) for a single subTestResult.

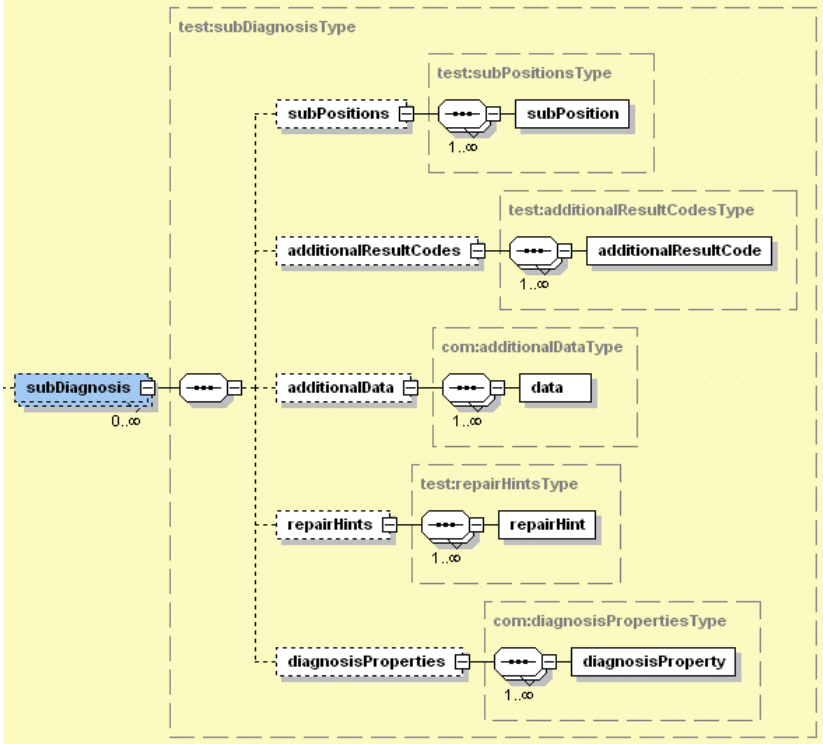


Fig 18: "control-1.1.xsd": Schema subDiagnosis

3.1.4.8.3 [repair]

The history of repair data.

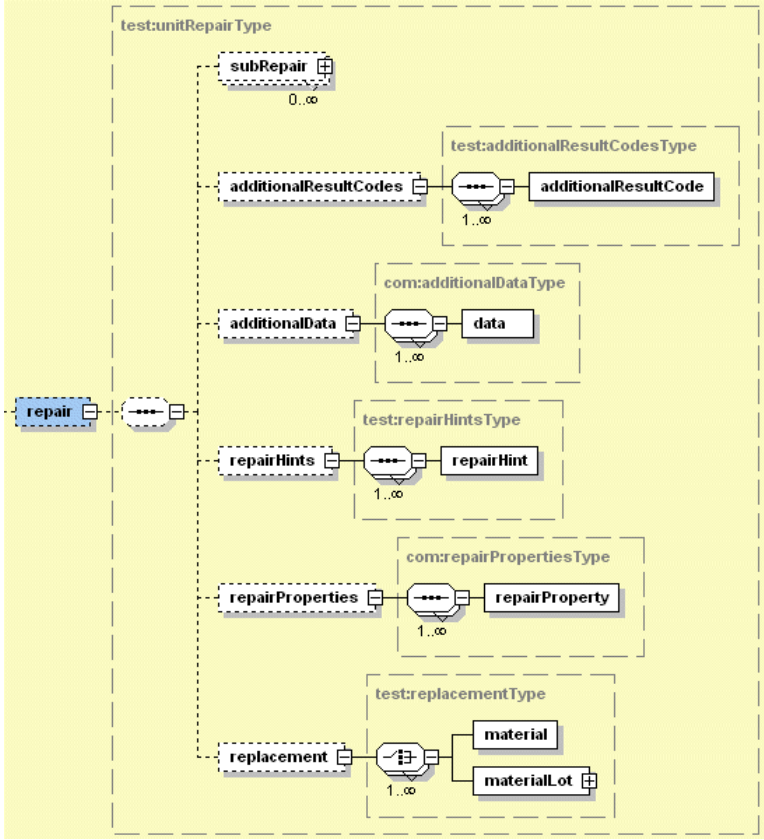


Fig 19: "control-1.1.xsd": Schema node testRepairHistory/repair

3.1.4.8.3.1 [subRepair]

Data for a repair performed on a product for a subTestResult.

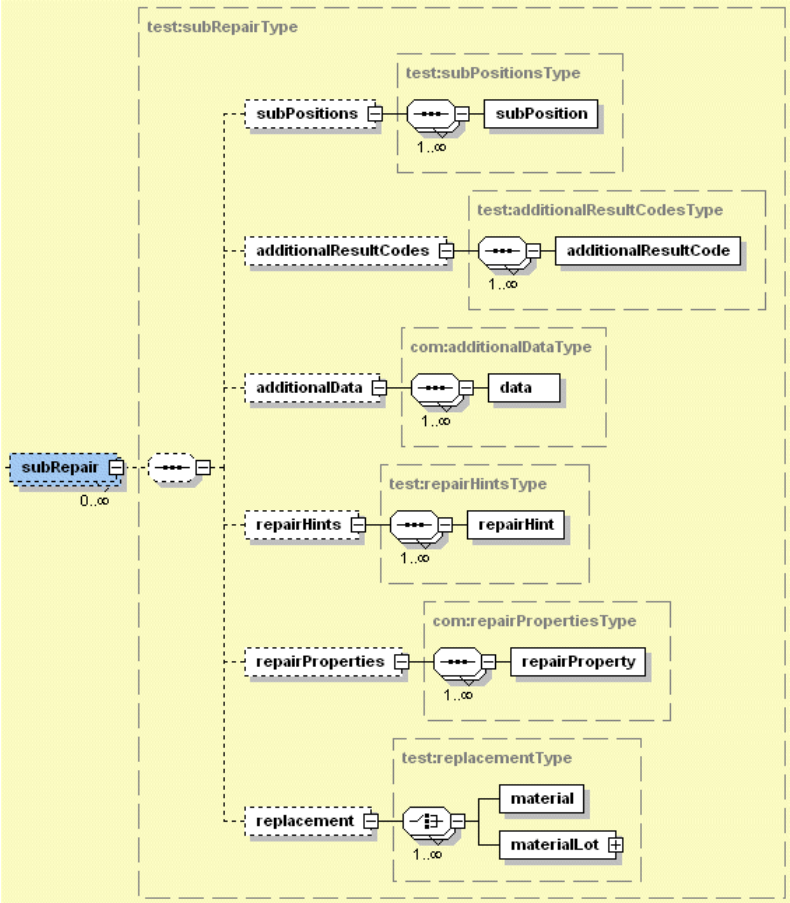


Fig 20: "control-1.1.xsd": Schema subRepair

3.1.4.9 [subUnitData]

Optional list of products (subassemblies) in a workpiece carrier.

Explanation!

Products can be combined in a workpiece carrier for shared processing. If the processing data of the individual products may differ, the data can be transferred individually for each product as a subUnitData. An example is the combination of independent circuits to a board (virtual workpiece carrier).

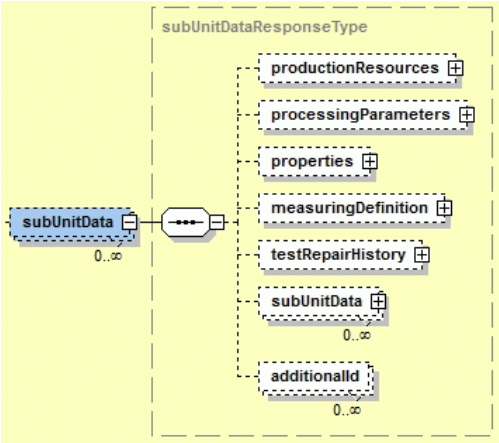


Fig 21: "control-1.1.xsd": Schema node subUnitData

3.1.4.9.1 [further subnodes]

Other optional subnodes with processing data may be located below the "subUnitData" node (similar to 3.1.4 [further subnodes])

- productionResources
- processingParameters
- properties
- measuringDefinition
- testRepairHistory
- subUnitData
- additionalId

3.1.4.10 [additionalId]

Optional list of additional serial numbers.

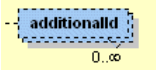


Fig 22: "control-1.1.xsd":: Schema additionalId

3.1.4.11 [additionalData]

Optional list of additional project-specific data.

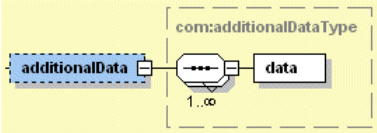


Fig 23: "control-1.1.xsd": Schema additionalData

3.1.4.12 [users]

Optional list of authenticated users or operators.

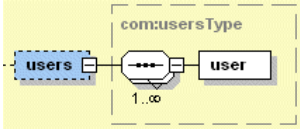


Fig 24: "control-1.1.xsd": Schema node users

3.1.4.13 [actions]

Optional list of actions to be performed.

Explanation!

To allow for concrete use of the interface, the structures between the communication partners must be co-ordinated in terms of content.

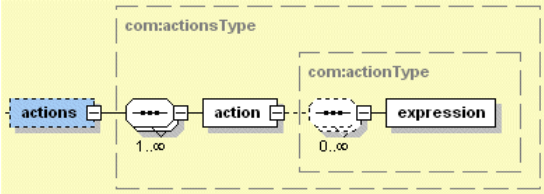


Fig 25: "control-1.1.xsd": Schema node actions

4 XML-Root: control

General properties of the XML root

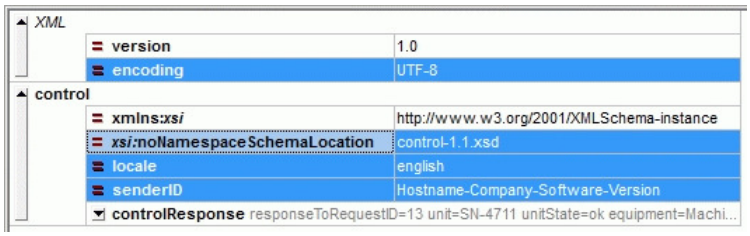


Fig 26: XML-Root control

Attribute	Format	Description
xmlns:xsi	URL	Link to the XMLSchema instance Example: http://www.w3.org/2001/XMLSchema-instance **Explanation:** No connection to the Internet is required.
xsi:noNamespaceSchemaLocation	XSD-File	Path (directory and name) of the XSD reference schema file **Explanation:** The reference schema file can be saved locally (same directory as the XML file) or in a general directory. If the reference schema file is saved in the same directory as the XML file, the directory does not need to be indicated. **Attention!** The path (directory and name) of the XSD reference schema file must be configurable in the sender's system.
[locale]	String	Optional language setting for messages and descriptions. **Explanation:** The language must be specified in English in lower case letters. Examples: german, english, spanish, chinese, french, hungarian, romanian.
[senderID]	String	Identification of the sender (eg computer name - company - software version). This attribute can be used to check the software version of the sender during software integration.

Table 1: XML-Root: /control/attribute::*

4.1 <controlRequest>

The requests for the preset data, which are to be used for processing a product, are handled by the 'controlRequest' subnode.

The requestID must be unique within the system. This can be achieved with a globally unique identifier (GUID) or with a timestamp counter.

The keys for a query may be the attributes "unit", "equipment", "equipmentClass", "operation" or "order". It is also possible to restrict the query by using a combination of these attributes.

In a control request, all requested data and attributes in the XML structure have to be present. Using these key data, in the answer the the results returnd are returned in the same XML structure.

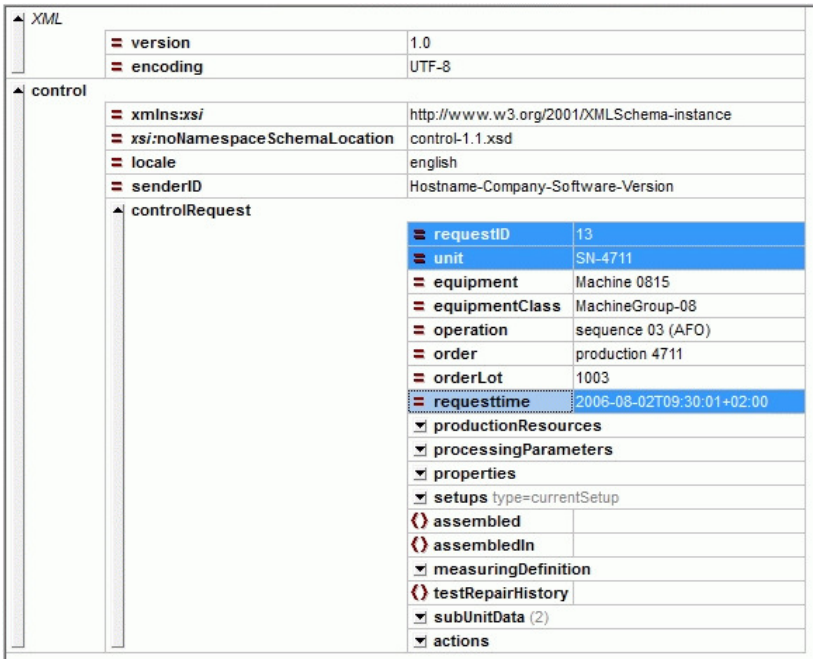


Fig 27: XML-Root /control/controlRequest

Attribute	Format	Description
requestID	**String**	**Unique identifier for the request; this may take the form of a counter.** **Explanation:** The uniqueness could be realised through the use of a timestamp. In this case, multiple requests must not be created within the smallest unit of time. The annual change from summer to winter time must thereby be taken into account.
[unit]	String	Serial number of the material being processed
[unitType]	String	Type of material being processed (for example Device, MaterialLot)
[unitSide]	String	Processed side of product (for example top or bottom side during testing)
[plant]	String	Plant, to restrict the query
[<equipment>]	String	Name of unique designation of the machine, manual workstation, production line or production cell
[<equipmentClass>]	String	Name of unique designation of a group of machines or manual workstations **Explanation:** This attribute can be used as an alternative to the "equipment" attributes in order to validate or exchange data for a group of machines or workstations.
[operation]	String	Name of the process operation that is to be performed.
[order]	String	Order number
[orderLot]	String	Lot number of the order
[material]	String	Material number of the manufactured product
[materialVersion]	String	Material version of the manufactured product
[materialVariant]	String	Material variant of the manufactured product
[operator]	String	Processor, machine technician
requesttime	**DateTime**	**Timestamp (date+time) when the request for the preset data occurred.**
[requestExpirationTime_s]	Numer	Specified time before the query expires, in seconds

Table 2: XML-Root: /control/controlRequest/attribute::*

4.2 <controlRequestCheck>

The requests for process interlocking are handled by the 'controlRequestCheck' subnode.

The requestID must be unique within the system. This can be achieved with a globally unique identifier (GUID) or with a timestamp counter.

The keys for a query may be the attributes "unit", "equipment", "equipmentClass", "operation" or "order". It is also possible to use a combination of these attributes in order to restrict the query.

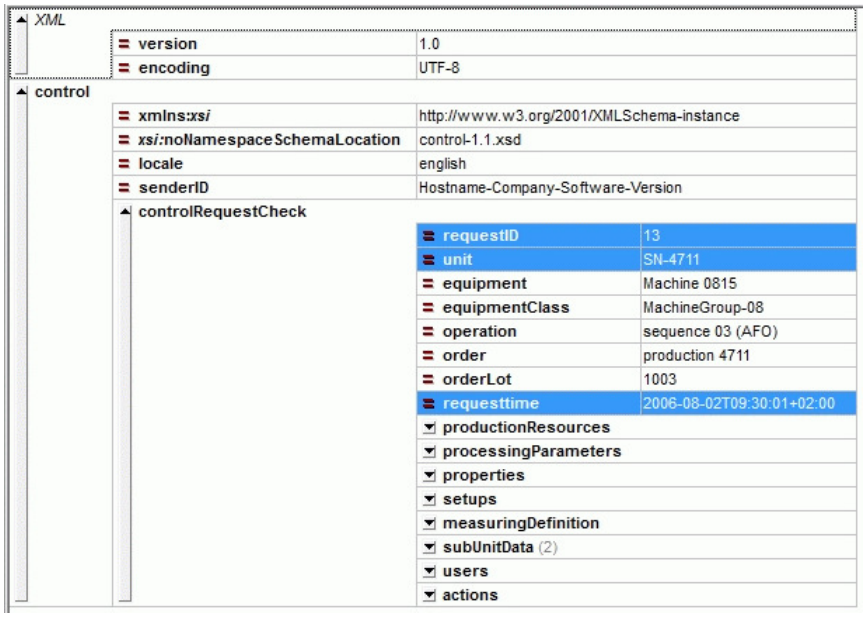


Fig 28: XML-Root /control/controlRequestCheck

Attribute	Format	Description
requestID	**String**	**Unique identifier for the request; this may take the form of a counter.** **Explanation:** The uniqueness could be realised through the use of a timestamp. In this case, multiple requests must not be created within the smallest unit of time. The annual change from summer to winter time must thereby be taken into account.
[unit]	String	Serial number of the material being processed
[unitType]	String	Type of material being processed (for example Device, MaterialLot)
[unitSide]	String	Processed side of product (for example top or bottom side during testing)
[plant]	String	Plant, to restrict the query
[<equipment>]	String	Name of unique designation of the machine, manual workstation, production line or production cell
[<equipmentClass>]	String	Name of unique designation of a group of machines or manual workstations **Explanation:** This attribute can be used as an alternative to the "equipment" attributes in order to validate or exchange data for a group of machines or workstations.
[operation]	String	Name of work step or process operation that is to be performed.
[order]	String	Order number
[orderLot]	String	Lot number of the order
[material]	String	Material number of the manufactured product
[materialVersion]	String	Material version of the manufactured product
[materialVariant]	String	Material variant of the manufactured product
[operator]	String	Processor, machine technician
requesttime	**DateTime**	**Timestamp (date+time) when the request for the preset data occurred.**
[requestExpirationTime_s]	Numer	Specified time before the query expires, in seconds

Table 3: XML-Root: /control/controlRequestCheck/attribute::*

4.3 <controlResponse>

The ‘controlResponse’ subnode contains the response to requests, which were sent using either the ‘controlRequest’ or the ‘controlRequestCheck’ nodes.

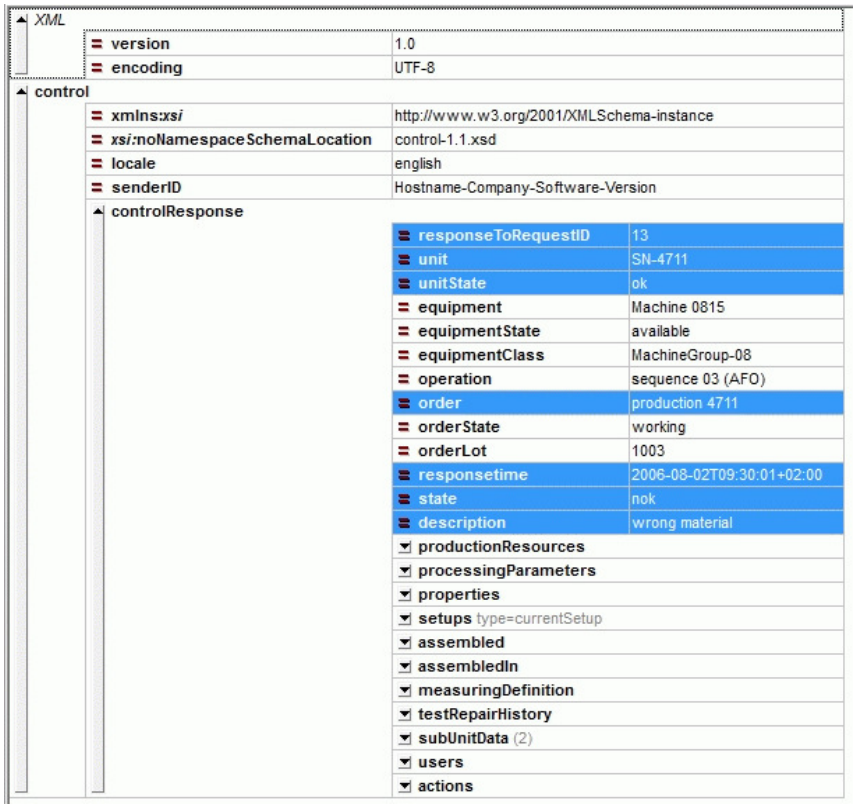


Fig 29: XML-Root /control/controlResponse

Attribute	Format	Description
responseToRequestID	**String**	**A replication of the unique identifier of the request.** **Explanation:** Using this identifier, multiple requests for a response can be uniquely assigned to a specific request.
[unit]	String	Serial number of the material being processed
[unitType]	String	Type of material being processed (for example Device, MaterialLot)
[unitSide]	String	Processed side of product (for example top or bottom side during testing)
[unitState]	String	Current status of the product that is to be processed
[unitProcessingState]	String	In addition to the attribute 'unitState' for the status of the product processing you can use the optional attribute 'unitProcessingState' to assign a process-related state. Example: unitState=ok, unitProcessingState=installed
[plant]	String	Plant in which the material was most recently processed
[<equipment>]	String	Unique identifier or number of the machine, the manual workstation, the production line or production cell
[equipmentState]	String	Current status of the machine, the manual workstation, the production line or production cell
[<equipmentClass>]	String	Name or unique descriptor for a group of machines or manual workstations. **Explanation:** This attribute can be used as an alternative to the "equipment" attributes in order to validate or exchange data for a group of machines or workstations.
[operation]	String	Name of the work step or process operation that is to be performed.
[order]	String	Order number.
[orderState]	String	Current status of the order that is to be processed
[orderLot]	String	Lot number of the order.
[material]	String	Material number of the manufactured product
[materialVersion]	String	Material version of the manufactured product
[materialVariant]	String	Material variant of the manufactured product
[operator]	String	Processor, machine technician
responsetime	**DateTime**	**Timestamp (date+time) when the response to the request for process release or process control occurred.**
state	**String**	**Result of the check on the process release.**
[description]	String	A detailed description of the error, this typically describes the cause of the negative result.

Table 4: XML-Root: /control/controlResponse/attribute::*

4.4 [further subnodes]

The alternative nodes "controlRequest", "controlRequestCheck" and "controlResponse" may optionally contain further subnodes. The schema for these subnodes differ in terms of attributes, depending on whether they are used within "controlRequest", "controlRequestCheck" or "controlResponse" node. The following pages contain descriptions of the subnodes and attributes in each case for the subnodes "controlRequest", "controlRequestCheck" and "controlResponse". The subnodes "controlRequestCheck" and "controlResponse" have essentially the same attributes for checking and return. "controlResponse" contains one additional attribute, namely "state" for reporting the status.

Example controlRequest

Attribute	Format	Description
"key"	**String**	**Keys for the query**

Table 5: node /control/controlRequest/example/attribute::*

Example controlRequestCheck, controlResponse

Attribute	Format	Description
"key"	**String**	**Keys for the query**
"value"	**String**	**Result of the query**
[state] only for controlResponse	String	Status of the result of the query

Table 6: node /control/controlResponse/example/attribute::*

4.4.1 [productionResources]

Tools and production resources that are to be used in the upcoming process step.

4.4.1.1 Resource

Listing of the production resources.

4.4.1.1.1 controlRequest

Fig 30: node /control/controlRequest/productionResources/resource

Attribute	Format	Description
[equipment]	String	Represents an optional means of assigning the tools and production resources to various sub areas within a single machine. **Explanation:** If equipment is not specified, the tool or production resource will be written to equipment in the 'controlRequest' node.
[position]	String	Optional specification of the position within a machine
type	**String**	**Type of production resource.**

Table 7: node /control/controlRequest/productionResources/resource/attribute::*

4.4.1.1.2 controlRequestCheck, controlResponse

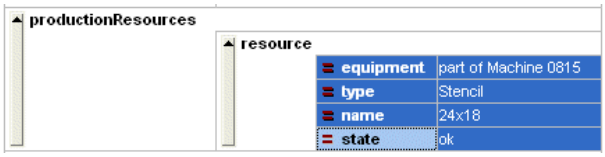


Fig 31: node /control/controlResponse/productionResources/resource

Attribute	Format	Description
[equipment]	String	Represents an optional means of assigning the tools and production resources to various sub areas within a single machine. **Explanation:** If equipment is not specified, the tool or production resource will be written to equipment in the 'controlResponse' node.
[position]	String	Optional specification of the position within a machine
type	**String**	**Type of production resource.**
name	**String**	**Name (instance) of production resource.**
[state] only for controlResponse	String	Detailed result regarding the check for process release.

Table 8: node /control/controlResponse/productionResources/resource/attribute::*

4.4.2 [processingParameters]

Parameters and nominal values that are to be used in the upcoming process step.

4.4.2.1 parameter

List of relevant production parameters, nominal values and set points.

4.4.2.1.1 controlRequest

processingParameters

parameter (3)

	equipment	name
1		Program
2		ProgramVersion
3	Stencil	Temperature

Fig 32: node /control/controlRequest/processingParameters/parameter

Attribute	Format	Description
[equipment]	String	Represents an optional means of assigning the tools and production resources to various sub areas within a single machine. **Explanation:** If equipment is not specified, the parameter, nominal value or set point will be written to equipment in the 'controlRequest' node.
[position]	String	Optional specification of the position within a machine to which a parameter or nominal value applies.
name	**String**	**Name of parameters or nominal values**

Table 9: node /control/controlRequest/processingParameters/parameter/attribute::*

4.4.2.1.2 controlRequestCheck, controlResponse

processingParameters

parameter (3)

	equipment	name	value	UnitOfMeasure	state
1		Program	Assemble-4711		ok
2		ProgramVersion	1.7		ok
3	Stencil	Temperature	320	K	ok

Fig 33: node /control/controlResponse/processingParameters/parameter

Attribute	Format	Description
[equipment]	String	Represents an optional means of assigning the tools and production resources to various sub areas within a single machine. **Explanation:** If equipment is not specified, the parameter, nominal value or set point will be written to equipment in the 'controlResponse' node.
[position]	String	Optional specification of the position within a machine to which a parameter or nominal value applies.
name	**String**	**Name des Parameters oder Sollwertes**
value	**String**	**Value of the parameter, nominal value or set point.**
[UnitOfMeasure]	String	Unit of the parameter or setpoint **Attention!** Always required for numeric parameters and setpoints. **Explanation:** The dimension must be entered as an SI unit of measure (see 5.4 List of relevant terms and abbreviations). In addition to SI units of measure, the following units of measure are also supported: pcs (pieces), ° (degrees), m² (square metres), l (litres), % (percent), db (decibels)
[measureDataType]	String	If the parameter or nominal value is numeric, the format of its value can be specified in this attribute. Possible formats: decimal, exponential, metricPrefix, hexadecimal, binary, string (default: decimal) (see 2.2.1.2.1 measureDataType)
[state] only for controlResponse	String	State of the parameter, ok or nok

Table 10: node /control/controlResponse/processingParameters/parameter/attribute::*

4.4.3 [properties]

Additional properties of objects.

Fig 34: node /control/node/properties

Explanation!

Properties can be passed for the objects “unit”, "equipment", "order", "material" or "operation".

- **unit:**
  Material that is to be processed
- **equipment:**
  Machine, manual workplace, production line or production cell
- **order**
  Production order
- **material**
  Manufactured product (article)
- **operation**
  Operation

4.4.3.1 <object>Property

List of object properties.

4.4.3.1.1 controlRequest

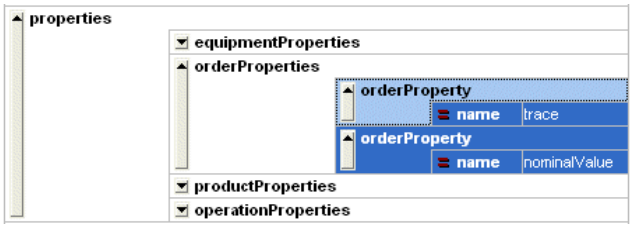


Fig 35: node /control/controlRequest/properties/property

Attribute	Format	Description
name	String	Type of property

Table 11: node /control/controlRequest/properties/property/attribute::*

4.4.3.1.2 controlRequestCheck, controlResponse

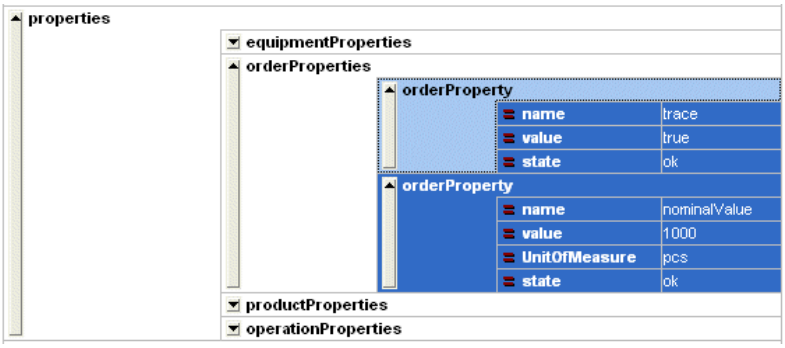


Fig 36: node /control/controlResponse/properties/property

Attribute	Format	Description
name	**String**	**Type of property**
value	String	**Value of the property**
[UnitOfMeasure]	String	Unit of measure of the property **Attention!** Always required for numeric parameters and setpoints. **Explanation:** The dimension must be entered as an SI unit of measure (see 5.4 List of relevant terms and abbreviations). In addition to SI units of measure, the following units of measure are also supported: pcs (pieces), ° (degrees), m² (square metres), l (litres), % (percent), db (decibels)
[state] only for controlResponse	String	Status of the parameter, ok or nok

Table 12: node /control/controlResponse/properties/property/attribute::*

4.4.4 [setups]

Setup data respectively materials, that are to be used in the upcoming process step.

4.4.4.1 Example for current setups without using setup equipments

In this case the node "setups" is used without using setup equipments (e.g. change over tabel, feeder).

4.4.4.1.1 controlRequest

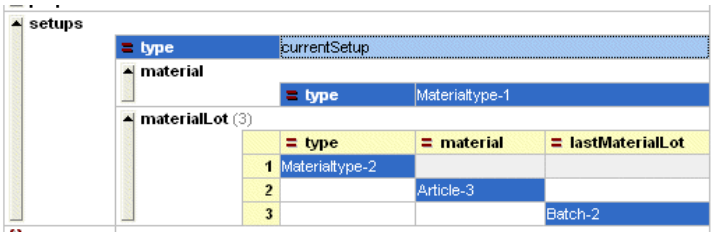


Fig 37: node /control/controlRequest/setups, currentSetup

Attribute	Format	Beschreibung
type	String	**nominalSetup** for requests concerning nominal setup data (e.g. materials to be set up) **currentSetup** requests concerning the current setup (e.g. the material lots, which currently are located at an equipment)
[name]	String	Optional name of a setup data set

Table 13: XML-Root: control/controlRequest/setups/attribute::*

Example node "material"

Attribute	Format	Description
type	String	Type of material

Table 14: node /control/controlRequest/setups/material/attribute::*

Example node "materialLot"

Attribute	Format	Description
<type>	String	Type of the material of a lot
<material>	String	Material number of the lot
<lastMaterialLot>	String	Unique identifier of a consumed lot, which requires replacement.

Table 15: node /control/controlRequest/setups/materialLot/attribute::*

Explanation!

A request must contain at least one of the attributes type, material or lastMaterialLot as key.

4.4.4.1.2 controlRequestCheck, controlResponse

Example node "material"

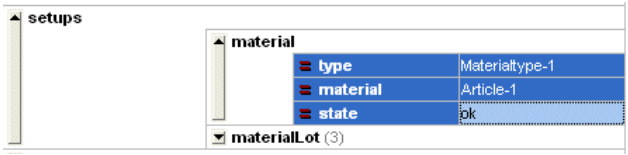


Fig 38: node /control/controlResponse/setups/material

Attribute	Format	Description
[type]	String	Type of the material
material	**String**	**Material number of the material, which needs to be assembled**
[state] only for controlResponse	String	Result of a sub query during process interlocking.

Table 16: node /control/controlResponse/setups/material/attribute::*

Example node "materialLot"

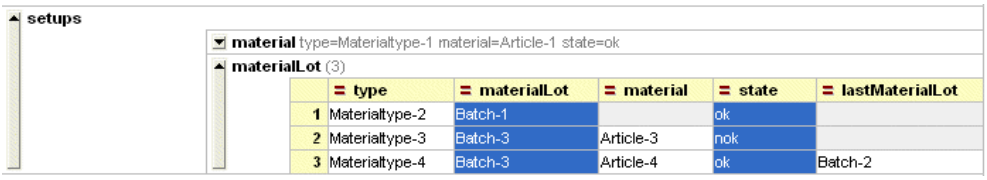


Fig 39: node /control/controlResponse/setups/materialLot

Attribute	Format	Description
[type]	String	Type of the material of a lot
materialLot	**String**	**Identifier of a lot of the material, which needs to be assembled.**
[material]	String	Material number of the lot
[lastMaterialLot]	String	Unique identifer of a consumed lot, which requires a replacement.
[state] only for controlResponse	String	Result of a sub query during process interlocking.

Table 17: node /control/controlResponse/setups/materialLot/attribute::*

4.4.4.2 Example for nominal setup data using setup equipments

This example explains the use of the node “setups” for nominal setup data, with usage of setup equipment like setup tables or feeders.

4.4.4.2.1 controlRequest

Fig 40: node /control/controlRequest/setups, nominalSetup

Attribute	Format	Beschreibung
type	String	**nominalSetup** for requests concerning nominal setup data (e.g. materials to be set up) **currentSetup** requests concerning the current setup (e.g. the material lots, which currently are located at an equipment)
[name]	String	Optional name of a setup data set

Table 18: node /control/controlRequest/setups/attribute::*

4.4.4.2.2 controlRequestCheck, controlResponse

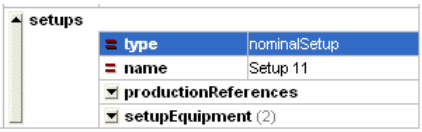


Fig 41: node /control/controlResponse/setups

Attribute	Format	Beschreibung
type	String	**nominalSetup** for requests concerning nominal setup data (e.g. materials to be set up) **currentSetup** requests concerning the current setup (e.g. the material lots, which currently are located at an equipment)
[name]	String	Optional name of a setup data set

Table 19: node /control/controlResponse/setups/attribute::*

4.4.4.2.3 [productionReferences]

Optional listing of orders, products or products sides, for which a setup data set is relevant.

productionReferences

productionReference (2)

	order	material	productSide
1	order-123		TOP
2		product-1	BOT

Fig 42: node /control/controlResponse/setups/productionReferences

Attribute	Format	Beschreibung
[order]	String	Order number, for which a setup data set should be used
[material]	String	Product number, for which a setup data set should be used
[productSide]	String	Product side, for which a setup data set should be used

Table 20: node control/controlResponse/setups/productionReferences/attribute::*

4.4.4.2.4 [setupEquipment]

The node setupEquipment can be used to model a tree of used setup equipment, like setup tables or feeders.

setupEquipment (2)

	position	positionName	type	material
1	1	FRONT_RIGHT	Table 10	material (3)
2	2	BACK_RIGHT	Table 10	material (3)

material (3)

	position	material
1	1.1	172559.0100
2	1.2	164814.0100
3	1.3	115912.0100

material (3)

	position	material
1	1.1	172559.0100
2	1.2	164814.0100
3	1.3	115912.0100

Fig 43: node /control/controlResponse/setups/setupEquipment

Attribute	Format	Beschreibung
[position]	String	Position of a setup equipment, e.g. of a table within a machine
[positionName]	String	Name of the position of a setup equipment
[type]	String	Type of a setup equipment

Table 21: node /control/controlResponse/setups/setupEquipment/attribute::*

4.4.4.2.4.1 [material]

Listing of materials, which are going to be added to the setup of a machine or setup equipment.

Attribute	Format	Beschreibung
[position]	String	optional position of a setup entry.
material	String	Material number

Table 22: node /control/controlResponse/setups/setupEquipment/material/attribute::*

4.4.5 [assembled]

Data about material consumptions and installed components.

4.4.5.1 controlRequest

Queries for data about assembled components.

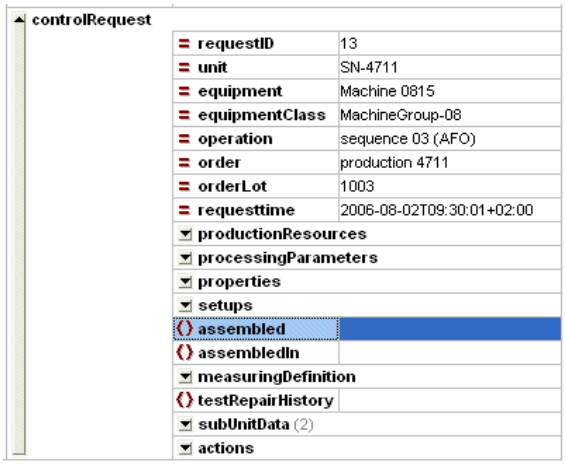


Fig 44: node /control/controlRequest/assembled

Explanation!

No attributes or subnodes are required for querying material consumption and installed material data with the "assembled" node.

Explanation!

The key for querying test and repair data is the serial number in the "unit" attributes of the "controlRequest" node.

4.4.5.2 controlRequestCheck

Checking the material consumption and installed material data is not provided..

4.4.5.3 controlResponse

The response nodes contain data about assembled components.

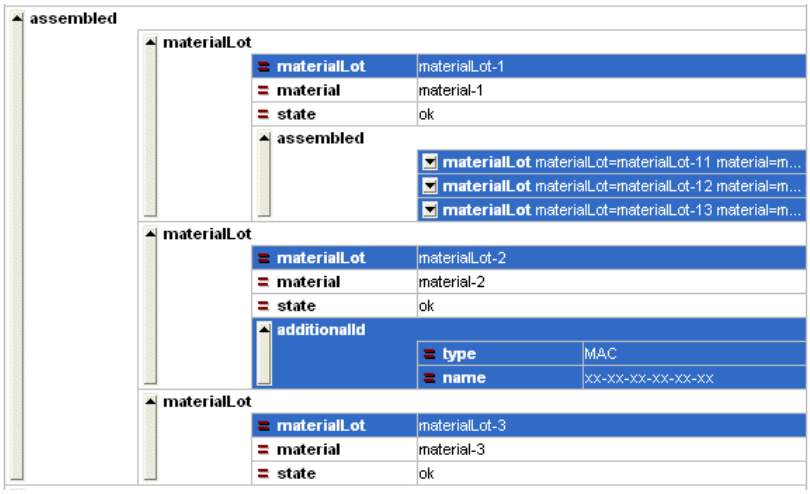


Fig 45: node /control/controlResponse/assembled

4.4.6 [assembledIn]

List of parent units where the processed material is installed.

4.4.6.1 controlRequest

Queries for the list of parent units where the processed material is installed.

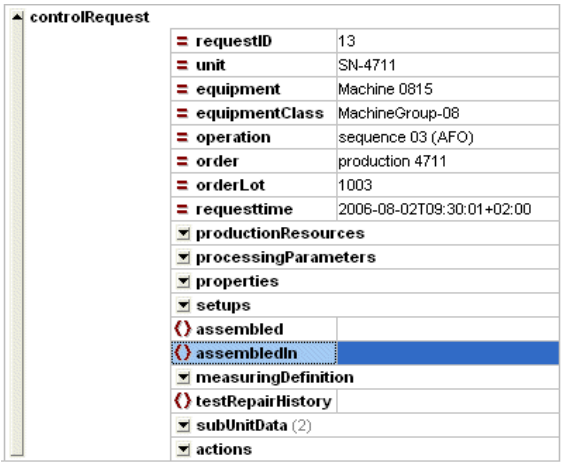


Fig 46: node /control/controlRequest/assembledIn

Explanation!

No attributes or subnodes are required for querying the list of parent units where the processed material is installed with the "assembled" node.

Explanation!

The key for querying the list of parent units where the processed material is installed is the serial number in the "unit" attributes of the "controlRequest" node.

4.4.6.2 controlRequestCheck

Checking the list of parent units where the processed material is installed is not provided..

4.4.6.3 controlResponse

The response nodes contain data about the parent units, in which the processed material is installed.

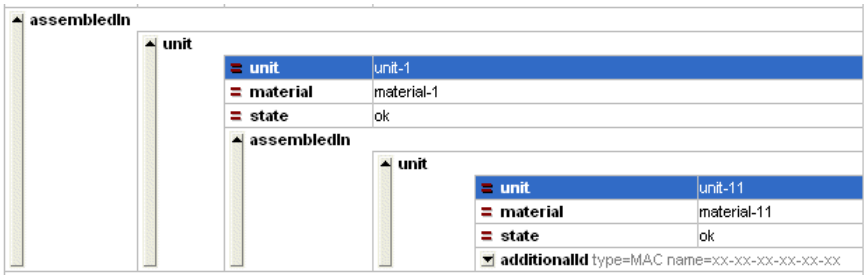


Fig 47: node /control/controlResponse/assembledIn

4.4.7 [measuringDefinition]

Nominal and boundary values for measurements.

4.4.7.1 channelDefinition

List of measurement channels

4.4.7.1.1 controlRequest

Fig 48: node /control/controlRequest/measuringDefinition/channelDefinition

Attribute	Format	Description
name	**String**	**Unique name of a measurement channel**

Table 23: node /control/controlRequest/measuringDefinition/channelDefinition /attribute::*

4.4.7.1.2 controlRequestCheck, controlResponse

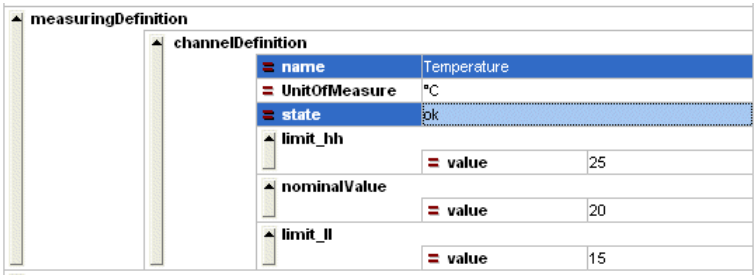


Fig 49: node /control/controlResponse/measuringDefinition/channelDefinition

Attribute	Format	Description
name	**String**	**Unique name of a measurement channel**
[UnitOfMeasure]	String	Unit of measurement for the channel **Attention!** Always required for numeric parameters and setpoints. **Explanation:** The dimension must be entered as an SI unit of measure (see 5.4 List of relevant terms and abbreviations). In addition to SI units of measure, the following units of measure are also supported: pcs (pieces), ° (degrees), m² (square metres), l (litres), % (percent), db (decibels)
[measureDataType]	String	If the measurement returned by a measurement channel is numeric, the format of its value can be specified in this attribute. Possible formats: decimal, exponential, metricPrefix, hexadecimal, binary, string Default: decimal
[state] only for controlResponse	String	Status of the parameter, ok or nok

Table 24: node /control/controlResponse/measuringDefinition/channelDefinition /attribute::*

The node channelDefinition is described in MIT-6 "ZVEI interface TestAndRepair", chapter "channelDefinition".

4.4.8 [testRepairHistory]

The history of test and repair data.

Explanation!

The history of test and repair data can be queried with the "controlRequest→testRepairHistory" node, but cannot validated with a "controlRequestCheck" node.

4.4.8.1 controlRequest

Querying the history of test and repair data.

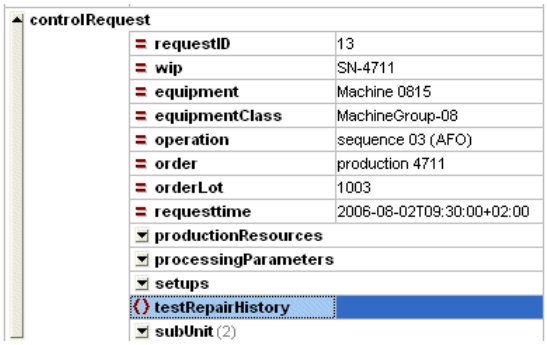


Fig 50: node /control/controlRequest/testRepairHistory

Explanation!

No attributes or subnodes are required for querying test and repair data with the "testRepairHistory" node.

Explanation!

The key for querying test and repair data is the serial number in the "unit" attributes of the "controlRequest" node. The query can be restricted here for a machine or a workstation ("controlReques/equipment" attributes) or a group of machines or workstations ("controlReques/equipmentClass" attributes).

4.4.8.2 controlRequestCheck

Checking the history of test and repair data is not provided..

4.4.8.3 controlResponse

Returning the history of test and repair data for a serial number.

Attention!

The test and repair nodes are listed in chronological order in the history. The timestamps of the data received from the test machines and workstations are used here.

Time synchronisation must take place between the communication partners. In order for this to occur, the plant operator must provide a suitable time server.

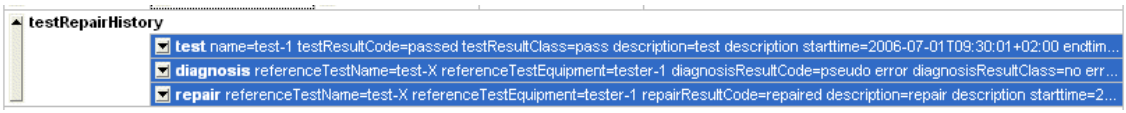


Fig 51: node /control/controlResponse/testRepairHistory

4.4.8.3.1 [test]

Test data that relates to a tested product.

Explanation!

Test data consists of one overall test and an arbitrary number of partial tests. A partial test has to be defined in the interface for the transfer of test details (e.g. the identifier of a test position).

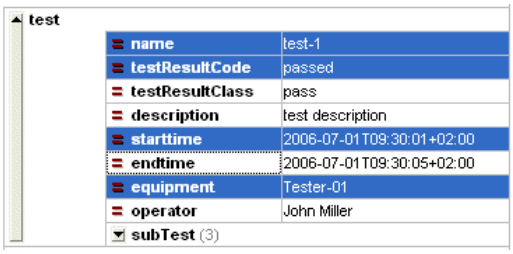


Fig 52: node /control/controlResponse/testRepairHistory/test

Attribute	Format	Description
name	**String**	**Unique name of a test or a test procedure.**
testResultCode	**String**	**Result of the overall test** **Explanation:** The codes for the possible test results should be configurable. Some examples of standard designations include: 'passed', 'failed', 'aborted'. The specific designation must be co-ordinated with the system operator.
[testResultClass]	String	Optional attribute for classification of test results. The following classes can be specified: 'pass', 'certifiedPass', 'fail', 'interrupt', 'unknown'. If no classification is transferred, the default 'unknown' is assumed.
[description]	String	Optional description of a test.
starttime	**DateTime**	**Timestamp (date+time) when processing of the product for this operation began.**
[endtime]	DateTime	Timestamp (date+time) when processing of the product for this operation ended.
equipment	**String**	**The test was carried out on this machine or workstation.**
[operator]	String	Person performing the test, person who adjusted the settings for the test machine.

Table 25: node /control/controlResponse/testRepairHistory/test/attribute::*

Explanation!

You will find further information on the "test" node in MIT-6 "ZVEI interface TestAndRepair", Section "test".

4.4.8.3.2 [diagnosis]

Data for diagnosing (analysis, classification) test data.

Explanation!

The transfer of diagnostic data requires a reference to the test for which the analysis or classification was carried out.

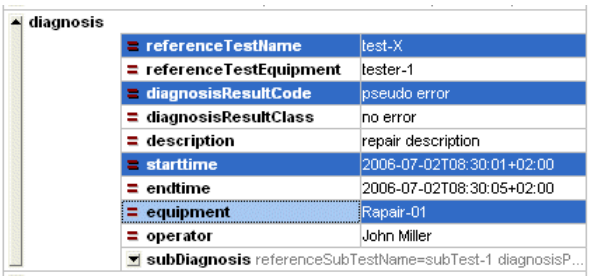


Fig 53: node /control/controlResponse/testRepairHistory/diagnosis

Attribute	Format	Description
referenceTestName	**String**	**Name of the test that triggered the repair.**
[referenceTestEquipment]	String	Name or unique descriptor of the test machine or test station whose test results triggered this repair.
diagnosisResultCode	**String**	**Result of the diagnosis** **Explanation:** The codes for the possible diagnosis results should be configurable. The specific designation must be co-ordinated with the system operator.
[diagnosisResultClass]	String	Optional attribute for classifying diagnostic results. The following classes can be specified: 'fault', 'pseudoFault', 'testFault', 'consecutiveFault', 'unknown'. When no classification is passed then the default 'unknown' applies.
[description]	String	Optional description or annotation to the diagnosis that has been performed.
starttime	**DateTime**	**Timestamp (date+time) when processing of the product for this operation began.**
[endtime]	DateTime	Timestamp (date+time) when processing of the product for this operation ended.
equipment	**String**	**Diagnosis was carried out on this machine or workstation.**
[operator]	String	Person performing the diagnosis

Table 26: node /control/controlResponse/testRepairHistory/diagnosis/attribute::*

Explanation!

You will find further information on the "diagnosis" node in MIT-6 "ZVEI interface TestAndRepair", Section "diagnosis".

4.4.8.3.3 [repair]

Data relating to a repair performed on a serial number.

Explanation!

The transfer of repair requires a reference to the test whose results triggered this repair.

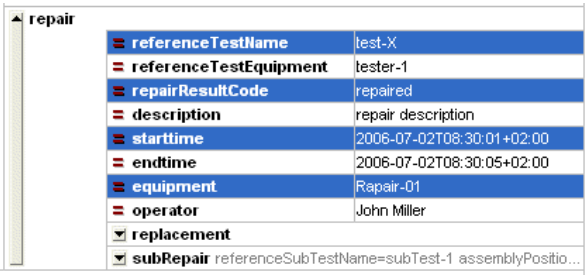


Fig 54: node /control/controlResponse/testRepairHistory/repair

Attribute	Format	Description
referenceTestName	**String**	**Name of the test whose result triggered this repair.**
[referenceTestEquipment]	String	Name or unique descriptor of the test machine or test station whose test results triggered this repair.
repairResultCode	**String**	**Result of the repair** **Explanation:** The codes for the possible repair results should be configurable. The specific designation must be co-ordinated with the system operator.
[description]	String	Optional description of or comment on the repair performed
starttime	**DateTime**	**Timestamp (date+time) when processing of the product for this operation began.**
[endtime]	DateTime	Timestamp (date+time) when processing of the product for this operation ended.
equipment	**String**	**Repair was carried out on this machine or workstation**
[operator]	String	Person performing the repair, person who adjusted the settings for the repair machine

Table 27: node /control/controlResponse/testRepairHistory/repair/attribute::*

Explanation!

Further details relating to the node "repair" are described in MIT-6 "ZVEI interface TestAndRepair", chapter "Repair".

4.4.9 [subUnitData]

Optional listing of products (sub-setups groups) in a workpiece holder.

Explanation!

Products can be combined in a workpiece carrier for shared processing. If the processing data of the individual products may differ, the data can be transferred individually for each product as a subUnitData. An example is the combination of independent circuits to a board (virtual workpiece carrier).

Explanation!

You must specify either the serial number of the processed product, or the serial number of the workpiece holder together with the (sequence) number within the workpiece holder.

4.4.9.1 controlRequest

subUnitData (2)

	subUnit	position	positionType
1	SN-4712	1	sequence
2	SN-4713	2	sequence

Fig 55: node /control/controlRequest/subUnitData

Attribute	Format	Description
<subUnit>	**String**	**Serial number of a processed product in a workpiece carrier** **Explanation:** Either the serial number of the processed product or the serial number of the workpiece carrier must be specified, combined with the number in the workpiece carrier (position).
[subUnitType]	String	Type of material being processed (for example Device, MaterialLot)
[subUnitSide]	String	Side of processed material (for example top or bottom side during testing)
<position>	**String**	**Position of the subproduct (subassembly) in the workpiece carrier**
<positionType>	**String**	**Type of the position in the workpiece carrier** • sequence • line,column • coordinates (x, y)

Table 28: node /control/controlRequest/subUnitData/attribute::*

4.4.9.2 controlRequestCheck, controlResponse

subUnitData (2)

	subUnit	position	positionType	state
1	SN-4712	1	sequence	ok
2	SN-4713	2	sequence	ok

Fig 56: node /control/controlResponse/subUnitData

Attribute	Format	Description
<subUnit>	**String**	**Serial number of a processed product in a workpiece carrier** **Explanation:** Either the serial number of the processed product or the serial number of the workpiece carrier must be specified, combined with the number in the workpiece carrier (position).
[subUnitType]	String	Type of material being processed (for example Device, MaterialLot)
[subUnitSide]	String	Side of processed material (for example top or bottom side during testing)
<position>	**String**	**Position of the subproduct (subassembly) in the workpiece carrier**
<positionType>	**String**	**Type of the position in the workpiece carrier** • sequence • line,column • coordinates (x, y)
[material]	String	Material number of the manufactured product
[materialVersion]	String	Material version of the manufactured product
[materialVariant]	String	Material variant of the manufactured product
[state] only for controlResponse	String	Result of the check on the process release for a subUnit.
[description] only for controlResponse	String	A detailed description of the error, this typically describes the cause of the negative result.

Table 29: node /control/controlResponse/subUnitData/attribute::*

4.4.9.3 [further subnodes]

Other optional subnodes with processing data may be located under the "subUnitData" node (similar to 4.4 [further subnodes])

- productionResources
- processingParameters
- properties
- measuringDefinition
- testRepairHistory
- subUnitData
- additionalId

4.4.10 [additionalId]

Optional list of additional serial numbers.

4.4.10.1 controlRequest, controlRequestCheck, controlResponse

Additional serial numbers can be used in all sub-nodes "controlRequest", "controlRequestCheck" and "controlResponse".

additionalId (1)

	type	name	state
1	CustomerID	ExtSN-12345678	assigned

Fig 57: node /control/controlResponse/additionalId

Attribute	Format	Description
type	**String**	**Type an additional serial number (e.g. customer serial number, equipment serial number)**
name	**String**	**additional serial number**
[state]	String	Optional status to an additional serial number (e.g. reserved or allocated) **Note:** The codes for the status should be configurable. Some examples are: 'ok', 'nok', 'aborted', 'BadBoard', 'repaired,' passed ',' failed '. The specific name is to be coordinated with the operator of the plant.

Table 30: node /control/controlResponse//additionalId/attribute::*

4.4.11 [additionalData]

Optional list of additional data.

Explanation!

This node can be used to transmit specific project data. The data itself or a link to the data (eg URL or file path) can be passed.

4.4.11.1 controlRequest, controlRequestCheck, controlResponse

AdditionalData can be used in "controlRequest", "controlRequestCheck" and "controlResponse".

4.4.11.1.1 data

Listing of specific project data

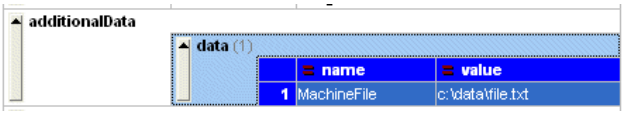


Fig 58: node /control/controlResponse/additionalData/data

Attribute	Format	Description
[type]	String	Type of data (eg. measurement, file, external source)
name	**String**	**Name**
value	**String**	**Value or reference (for example URL or file path)**

Table 31: node /control/controlResponse/additionalData/data/attribute::*

4.4.12 [users]

Optional list of authenticated users (users, operators).

4.4.12.1 controlRequest

There is no provision for queries relating to users.

4.4.12.2 controlRequestCheck, controlResponse

You can use controlRequestCheck and controlResponse to check the validity of an account in an external system.

4.4.12.2.1 user

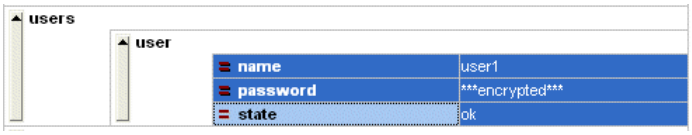


Fig 59: node /control/controlResponse/users/user

Attribute	Format	Description
name	**String**	**Name of an authenticated user (user, operator)**
[password] only for controlRequestCheck	String	Password of an authenticated user (user, operator) **Explanation:** The password is passed in an encrypted form. This encryption applies only to the data transfer.
[state] only for controlResponse	String	Status of the user, ok (authenticated) or nok

Table 32: node /control/controlResponse/users/user/attribute::*

4.4.13 [actions]

Optional list of actions to be performed.

Explanation!

To allow for concrete use of the interface, the structures between the communication partners must be co-ordinated in terms of content.

4.4.13.1 controlRequest, controlRequestCheck, controlResponse

Actions to be performed can be specified in any of the partial nodes "controlRequest", "controlRequestCheck" and "controlResponse".

4.4.13.1.1 action

An action to be performed.

Fig 60: node /control/node/actions/action

Attribute	Format	Description
name	String	An action to be done

Table 33: node /control/node/actions/action/attribute::*

4.4.13.1.1.1 [expression]

Optional list of expressions to restrict or more narrowly define an action to be performed. The expression may be a simple string used to describe an object or a regular expression.

Explanation!

A regular expression (abbreviated as regex or regexp) is a special text sequence used to describe a search pattern (similar to a wildcard).
Wildcards such as *.txt are well known in file management.

The regex equivalent to *.txt is `(?i)\.txt$` where
- (?i) means ("ignore case")
- `$ means` „End of line“ or „end of string“ - depending on the context

Additional information about regular expressions is available on the Internet, for example from http://www.regular-expressions.info .

Attribute	Format	Description
name	String	Designation of an expression or regular expression.

Table 34: node /control/node/actions/action/expression/attribute::*

4.4.13.2 Examples

The following table shows some examples of the use of the node actions.

Action	Expression	Beschreibung
reserve	unit	Pull Unit SN
reserve	additionalId	Pull additional SN
GetAssembled	allLevels	Body structure query - only "Units" and no restriction possible
GetAssembledIn	allLevels	Installation location query – recursive query and no restriction possible
GetSubWipState	-	Status of all SubUnits
Setup	-	Arm
Desetup	allComponents	Disarm all armed Packagings
Desetup	notRequestedComponents	Disarm all packages that are not included in the request (combined arm + disarm)
Login	-	user login
Logout	-	user logout

5 Appendix

5.1 List of relevant documents

MIT-1 "Guideline for Identification and Traceability"	ZVEI manual for the entire supply and value-added chain
MIT-2 "ZVEI-Interfaces-ChangeHistory"	This document describes the history of changes to the interfaces control and unitData.
MIT-3 "ZVEI-Interfaces-Overview"	This document provides an overview of possible applications of the ZVEI control and unitData interfaces during the manufacturing of a product
MIT-4 "ZVEI interface control"	This document describes the structure of the standard ***control*** interface for transferring data (requests and return messages) in process control (advanced process control) during the processing of a product.
MIT-5 "ZVEI interface unitData"	This document describes the structure of the standard ***unitData*** interface for transferring data to process a product.
MIT-6 "ZVEI interface TestAndRepair"	Description of the substructures of ZVEI standard interfaces for transferring test and repair data
MIT-7 "ZVEI interface transfer protocols"	Description of transfer protocols for transferring XML structures for ZVEI standard interfaces
MIT-8 "control-1.1.xsd"	XSD schema as file for ZVEI standard interface control
MIT-9 "unitData-1.1.xsd"	XSD schema as file for ZVEI standard interface unitData
MIT-10 "ZVEI-common-1.1.xsd"	XSD schema as file for general types that are used in various ZVEI standard interfaces
MIT-11 "ZVEI-testRepair-1.1.xsd"	XSD schema as file for general types used to forward test and repair data that are used in various ZVEI standard interfaces
MIT-12 "control_Request-1.1.xml"	Sample request for the ZVEI standard control interface as a file
MIT-13 "control_Response-1.1.xml"	Sample return message for the ZVEI standard control interface as a file
MIT-14 "unitData-1.1.xml"	Example of the ZVEI standard unitData interface as a file

5.2 List of Figs

Fig 1: Example of an XPath ... 3
Fig 2: XML encoding ... 4
Fig 3: "control-1.1.xsd": Schema ... 8
Fig 4: "control-1.1.xsd": Schema controlRequest ... 9
Fig 5: "control-1.1.xsd": Schema controlRequestCheck ... 9
Fig 6: "control-1.1.xsd": Schema controlResponse ... 10
Fig 7: "control-1.1.xsd": Schema node productionResources ... 11
Fig 8: "control-1.1.xsd": Schema node processingParameters ... 11
Fig 9: "control-1.1.xsd": Schema node properties ... 12
Fig 10: "control-1.1.xsd": Schema node setups ... 13
Fig 11: "control-1.1.xsd": Schema node assembled ... 13
Fig 12: "control-1.1.xsd": Schema node assembledIn ... 13
Fig 13: "control-1.1.xsd": Schema node measuringDefinition ... 14
Fig 14: "control-1.1.xsd": Schema node testRepairHistory ... 14
Fig 15: "control-1.1.xsd": Schema node testRepairHistory/test ... 15
Fig 16: "control-1.1.xsd": Schema test/subTest/subTestResult ... 16
Fig 17: "control-1.1.xsd": Schema node testRepairHistory/diagnosis ... 17
Fig 18: "control-1.1.xsd": Schema subDiagnosis ... 18
Fig 19: "control-1.1.xsd": Schema node testRepairHistory/repair ... 19
Fig 20: "control-1.1.xsd": Schema subRepair ... 20
Fig 21: "control-1.1.xsd": Schema node subUnitData ... 21
Fig 22: "control-1.1.xsd":: Schema additionalId ... 22
Fig 23: "control-1.1.xsd": Schema additionalData ... 22
Fig 24: "control-1.1.xsd": Schema node users ... 22
Fig 25: "control-1.1.xsd": Schema node actions ... 22
Fig 26: XML-Root control ... 23
Fig 27: XML-Root /control/controlRequest ... 24
Fig 28: XML-Root /control/controlRequestCheck ... 26
Fig 29: XML-Root /control/controlResponse ... 28
Fig 30: node /control/controlRequest/productionResources/resource ... 31
Fig 31: node /control/controlResponse/productionResources/resource ... 31
Fig 32: node /control/controlRequest/processingParameters/parameter ... 32
Fig 33: node /control/controlResponse/processingParameters/parameter ... 33
Fig 34: node /control/node/properties ... 34
Fig 35: node /control/controlRequest/properties/property ... 34
Fig 36: node /control/controlResponse/properties/property ... 35
Fig 37: node /control/controlRequest/setups, currentSetup ... 36
Fig 38: node /control/controlResponse/setups/material ... 37
Fig 39: node /control/controlResponse/setups/materialLot ... 37
Fig 40: node /control/controlRequest/setups, nominalSetup ... 38
Fig 41: node /control/controlResponse/setups ... 38
Fig 42: node /control/controlResponse/setups/productionReferences ... 39
Fig 43: node /control/controlResponse/setups/setupEquipment ... 39
Fig 44: node /control/controlRequest/assembled ... 40
Fig 45: node /control/controlResponse/assembled ... 40
Fig 46: node /control/controlRequest/assembledIn ... 41
Fig 47: node /control/controlResponse/assembledIn ... 41
Fig 48: node /control/controlRequest/measuringDefinition/channelDefinition ... 42
Fig 49: node /control/controlResponse/measuringDefinition/channelDefinition ... 42
Fig 50: node /control/controlRequest/testRepairHistory ... 43
Fig 51: node /control/controlResponse/testRepairHistory ... 44
Fig 52: node /control/controlResponse/testRepairHistory/test ... 45
Fig 53: node /control/controlResponse/testRepairHistory/diagnosis ... 46
Fig 54: node /control/controlResponse/testRepairHistory/repair ... 47
Fig 55: node /control/controlRequest/subUnitData ... 48
Fig 56: node /control/controlResponse/subUnitData ... 49
Fig 57: node /control/controlResponse/additionalId ... 50
Fig 58: node /control/controlResponse/additionalData/data ... 51
Fig 59: node /control/controlResponse/users/user ... 52

Fig 60: node /control/node/actions/action 53

5.3 List of tables

Table 1: XML-Root: /control/attribute::* ... 23
Table 2: XML-Root: /control/controlRequest/attribute::* ... 25
Table 3: XML-Root: /control/controlRequestCheck/attribute::* ... 27
Table 4: XML-Root: /control/controlResponse/attribute::* ... 29
Table 5: node /control/controlRequest/example/attribute::* ... 30
Table 6: node /control/controlResponse/example/attribute::* ... 30
Table 7: node /control/controlRequest/productionResources/resource/attribute::* ... 31
Table 8: node /control/controlResponse/productionResources/resource/attribute::* ... 31
Table 9: node /control/controlRequest/processingParameters/parameter/attribute::* ... 32
Table 10: node /control/controlResponse/processingParameters/parameter/attribute::* ... 33
Table 11: node /control/controlRequest/properties/property/attribute::* ... 34
Table 12: node /control/controlResponse/properties/property/attribute::* ... 35
Table 13: XML-Root: control/controlRequest/setups/attribute::* ... 36
Table 14: node /control/controlRequest/setups/material/attribute::* ... 36
Table 15: node /control/controlRequest/setups/materialLot/attribute::* ... 36
Table 16: node /control/controlResponse/setups/material/attribute::* ... 37
Table 17: node /control/controlResponse/setups/materialLot/attribute::* ... 37
Table 18: node /control/controlRequest/setups/attribute::* ... 38
Table 19: node /control/controlResponse/setups/attribute::* ... 38
Table 20: node control/controlResponse/setups/productionReferences/attribute::* ... 39
Table 21: node /control/controlResponse/setups/setupEquipment/attribute::* ... 39
Table 22: node /control/controlResponse/setups/setupEquipment/material/attribute::* ... 39
Table 23: node /control/controlRequest/measuringDefinition/channelDefinition /attribute::* ... 42
Table 24: node /control/controlResponse/measuringDefinition/channelDefinition /attribute::* ... 42
Table 25: node /control/controlResponse/testRepairHistory/test/attribute::* ... 45
Table 26: node /control/controlResponse/testRepairHistory/diagnosis/attribute::* ... 46
Table 27: node /control/controlResponse/testRepairHistory/repair/attribute::* ... 47
Table 28: node /control/controlRequest/subUnitData/attribute::* ... 48
Table 29: node /control/controlResponse/subUnitData/attribute::* ... 49
Table 30: node /control/controlResponse//additionalId/attribute::* ... 50
Table 31: node /control/controlResponse/additionalData/data/attribute::* ... 51
Table 32: node /control/controlResponse/users/user/attribute::* ... 52
Table 33: node /control/node/actions/action/attribute::* ... 53
Table 34: node /control/node/actions/action/expression/attribute::* ... 53

5.4 List of relevant terms and abbreviations

Term	Description
Product	Definition according to GPSG / ProdHG Description of a material that is produced Examples: • Electronic flat module • Mechanical assembly • Device
Parts list	Description of the type and number of components included in a product
Work sequence	Description of a work plan with the production steps necessary to manufacture a product.
Work process	Description of an individual production step (work or process step) that is required as part of the work sequence to manufacture a product. A work process does not need to be assigned yet to any specific equipment.
Equipment	Resources used to manufacture a product such as • Production machines • Test machines • Manual workstations • Production lines • Production cells • Handling systems • Scanners and readers • Terminals (GUI) • Signal devices (traffic signals)
PCB	Unfitted independent circuit
Flat module	Fitted independent circuit
Panel	A combination of independent circuits to form a board (virtual workpiece carrier); single or multiple panels depending on the number of independent circuits
Panel blank	Unprocessed single or double panel
Serial number	Unique number of a manufactured part
Component	A component (component, assembly, bulk material) is used for a product (for example mounted, fitted)
Batch	Physical combination of individual parts, for example roller, rod and tray in a smallest packaging unit.
Lot	Quantity of parts that have a barcode (batch). There are two types of lots, supplier lot and incoming goods lot. Supplier lots can be subdivided into smaller incoming goods lots.
MaterialLot (lot number or batch number)	The unique number of a lot or batch. The number of a supplier lot is already assigned by the supplier at delivery. The numbers of incoming goods lots can be (re)assigned in incoming goods. If incoming goods lots are combined in a smallest packaging unit it is equivalent to a batch.
SI unit	Abbreviation for: "Système International d'Unités" SI units are part of the International System of Units for Natural Sciences. This is the standard system as defined by law in most countries. It describes seven fundamental SI units: • Length: Metre (m) • Weight: Kilogram (kg) • Time: Second (s) • Current strength: Ampere (A) • Temperature: Kelvin (K) • Amount of substance: Mole (mol) • Light intensity: Candela (cd) Numerous other units are derived from these basic units, for example Newton, Pascal or Joule.