

MOIS Procedure XML Schema ICD

Document version: Issue 1.8

DOCUMENT STATUS SHEET

Date	Version	Author	Reason for change
12/02/2011	1.0	S. Varadarajulu	
09/10/2012	1.0r2	K. Yeung	Changed to import and export
26/03/2013	1.1	W. Heinen	ESOC Additions
04/09/2013	1.2	W. Heinen	Industry Update
25/03/2015	1.3	W. Heinen	DLR: relax constraint on empty steps
27/11/2015	1.4	W. Heinen	DLR: added fixed attribute for TCPs
21/02/2020	1.5	D. Diaz	ESOC: XML GOTO format for EUD
15/05/2023	1.6	M. Renker	ESOC: seqGroupId PUS-C compliant & tlmDisplayRef
13/11/2023	1.7	M. Renker	ESOC: seqFlagSchedule
13/11/2023	1.8	M. Renker	ESOC: FREETEXT

Table of Contents

1.	Abstract	4
2.	Introduction.....	4
2.1	Purpose.....	4
2.2	Scope.....	4
3.	Definitions and Acronyms	4
3.1	Terminology.....	4
3.2	Acronyms.....	4
4.	Structural Overview	5
4.1.1	Structure.....	6
4.1.2	Variables	10
4.1.3	Set Variable Statement.....	11
4.1.4	Command Flags	11
4.1.5	Radix Enumeration	11
4.1.6	Time Type.....	12
4.1.7	Directive Call Statement.....	12
5.	Data Integrity Checks Performed on Import.....	13
	APPENDIX 1 Schema Proc.xsd	14
	APPENDIX 2 DOCUMENT CHANGE RECORD	270

1. ABSTRACT

This is the interface control document for a MOIS procedure. It is fully defined by the annotated XML schema documented in Appendix A (also available electronically as .xsd). MOIS procedures can be executed automatically or manually.

2. INTRODUCTION

2.1 PURPOSE

This document defines the structure of a MOIS procedure in XML. This is intended as an interchange format which MOIS imports into an internal dynamic representation and can export again after procedure editing. It will be used as an intermediate format when exporting Operations Language scripts such as STOL in order to have language adapters independent of the core MOIS.

MOIS can be used to write procedures against a number of supported spacecraft database formats relating to different control systems. We also have a parallel XML standard for spacecraft database exchange, called DETOX.

2.2 SCOPE

The definition is generic and not tied to a standard such as the CCSDS TDM frames or ECSS PUS.

3. DEFINITIONS AND ACRONYMS

3.1 TERMINOLOGY

The term ‘raw value’ is used to indicate a parameter value prior to calibration (for monitoring parameters) or resulting from a de-calibration (for command/sequence parameters). The term ‘engineering value’ is used to indicate a parameter’s calibrated value.

3.2 ACRONYMS

CCSDS	Consultative Committee for Space Data Systems
DETOX	Database Exchange of TM/TC for Operations XML
ECSS	European Cooperation for Space Standardization
ICD	Interface Control Document
ID	Identifier such as a relational key
PUS	Packet Utilisation Standard
MOIS	Manufacturing and Operations Information System
TC	Telecomamnd

STOL	Spacecraft Test and Operations Language
TDM	Time Domain Multiplexed (frames)
TM	Telemetry

4. STRUCTURAL OVERVIEW

A procedure has 3 container classes: the procedure itself (PROC), steps within the procedure (STEP) and statements within steps (STMT).

Structure is defined at the step level with flow-chart paths (PATH) between steps. Program structure can be inferred from this (if sufficiently well-formed).

Procedure parameters as well as local and global variables (VARIABLES) are defined at the level of the procedure. Certain elements within statements can refer to these variables.

Figure 1 shows the overall structure of the schema. For clarity, not all low-level elements are shown.

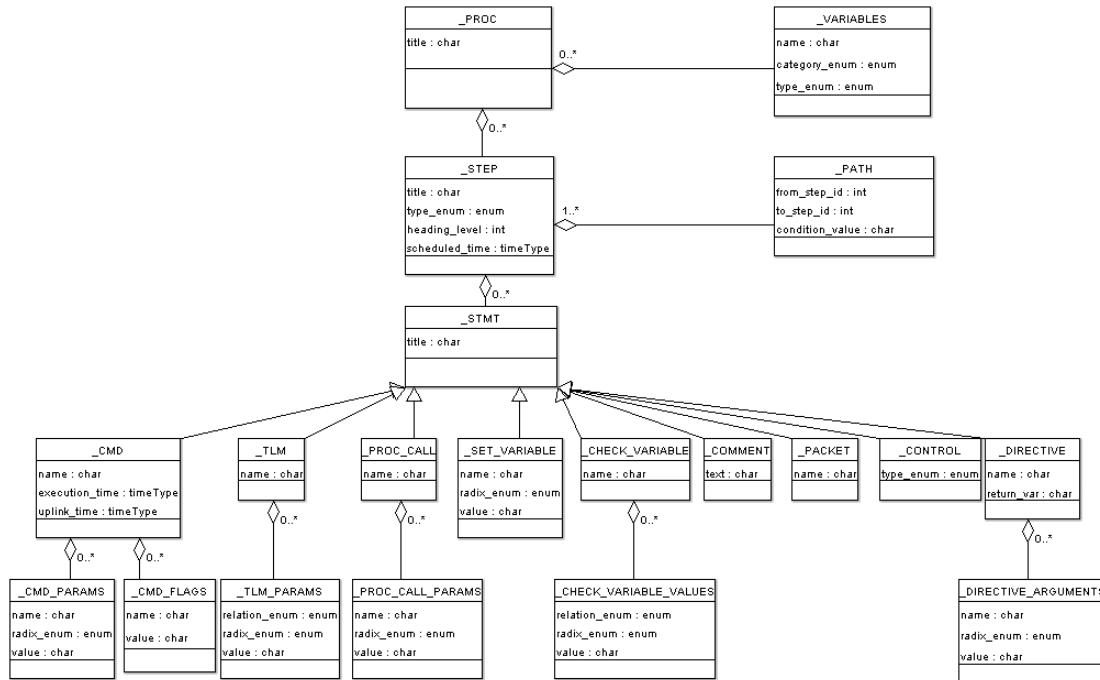


Figure 1 Procedure Overview

4.1.1 STRUCTURE

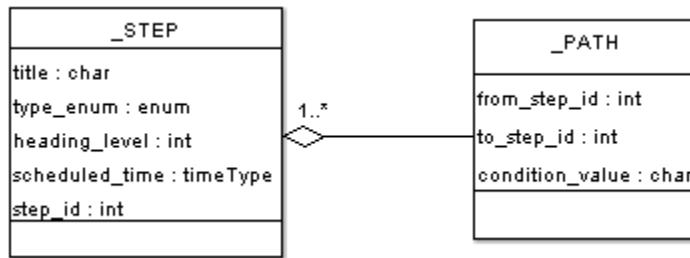


Figure 2 Structure representation in the Schema

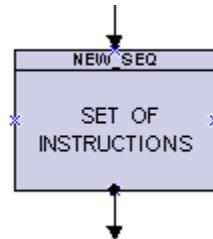
Structure is represented using flow-chart like connectors between steps, defined by PATH element entries; each entry identifies the 2 steps it is attached to. Condition steps are labelled (true, false, timeout etc.) with the condition_value element. A flow-chart representation can be converted to a structured Operations Language script as long as it is well-formed for the purpose. In other words as long as there is no external branching into loops (something that can only be achieved using GoTos and Labels in the code). This condition is not enforced by the schema. If the flow-chart contains external branching, then all steps are exported as PERFORM steps in a linear way as presented to the user in the tabular view of procedure body. The flowchart branching information is exported as GoTo labels within the step title (format: "[<branch condition label> -> <step number>]" e.g. "[yes -> 3 , no -> 4.1]" or "[case1 -> 23, case2 -> 24, any_other -> 50]").

Condition steps (all except the PERFORM step) must contain at least 1 Boolean TLM or CHECK_VARIABLE statement. If there is more than one a Boolean condition is required in the step_expression element of the step. An example is “([w] && [x]) || ([y] && [z])” where w, x, y and z are the corresponding stmt_title entries for the TLM and CHECK_VARIABLE statements.

The following step types (the type_enum element in Figure 2 above) are defined below (STARTPROC and ENDPROC are not included as these are virtual marker steps for the PATH table).

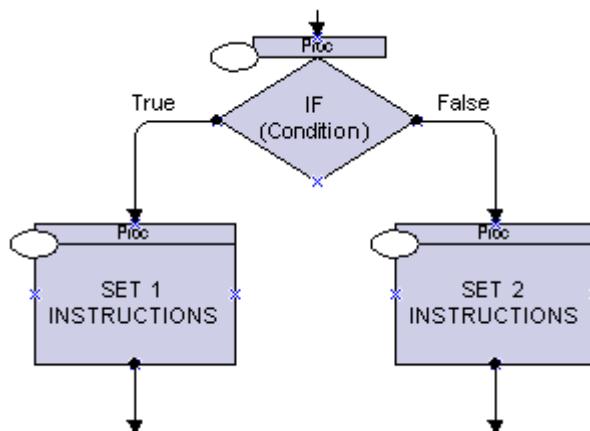
PERFORM

A Perform step is a step of any level (heading_level in Figure 2 above) in that is expected to consist of sub-steps or statements executing sequential instructions. Schematically:



IF

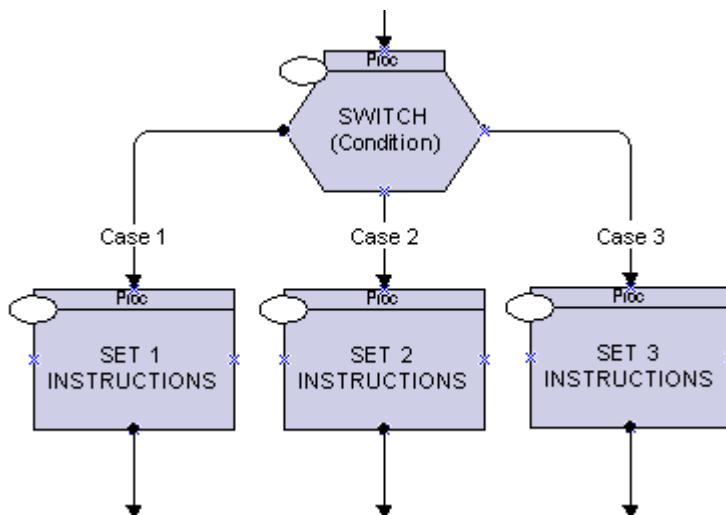
An If step defines a condition to be checked. Depending on whether this condition is true or false, one of two possible sets of instructions is executed. Each instruction set must begin with a level 1 step. Schematically:



The If step must contain statements of type TLM or CHECK_VARIABLE which define the condition check via a Boolean expression relating these Boolean statements.

SWITCH

The Switch step is a level 1 step that tests on three or more conditions and performs a corresponding set of instructions. Schematically:

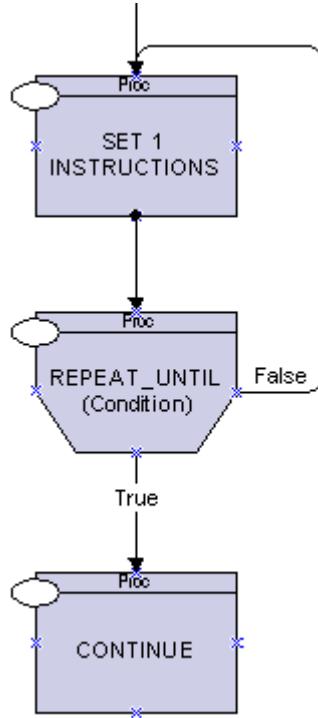


When checking the conditions, the first one to be satisfied (reading left to right in the above diagram) is implemented and the others ignored, even if they would also have been satisfied. If none of the defined conditions is satisfied, execution is halted and it is up to the user to resume it.

The initial statement corresponding to each case condition must contain statements of type TLM or CHECK_VARIABLE which define the condition check via a Boolean expression relating these Boolean statements. For all but the last case it is compulsory to have a condition specified. For the last case a condition is optional - it will be treated as an ELSEIF if a condition is specified, or an ELSE if there is no condition.

REPEAT_UNTIL Loop

The Repeat Until step is a level 1 step that performs a set of instructions until a specified condition is fulfilled. Schematically:

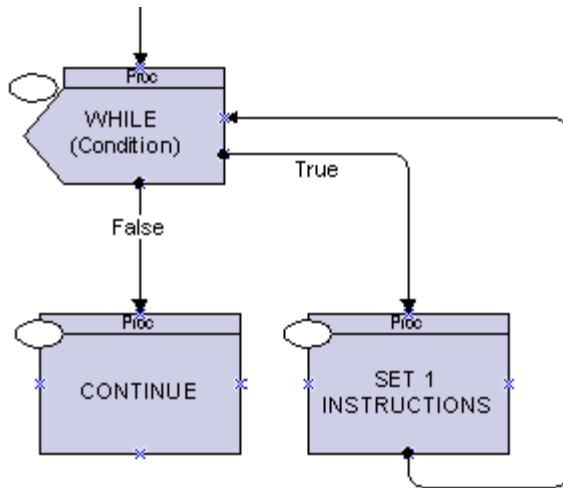


Note that the control check is placed at the *end* of the loop. The loop is therefore executed *at least once* even if the condition is false. The loop branch must point to a top-level step that is placed before the Repeat Until step.

The Repeat Until step must contain statements of type TLM or CHECK_VARIABLE which define the condition check via a Boolean expression relating these Boolean statements.

WHILE Loop

The While step is a level 1 step that performs a set of instructions as long as a specified condition is fulfilled. Schematically:



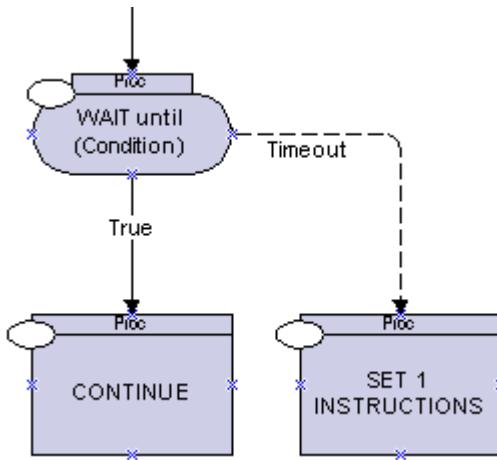
Note that the control check is placed at the *beginning* of the loop

The loop branch must point to a top-level step that is placed after the While step.

The While step must contain statements of type TLM or CHECK_VARIABLE which define the condition check via a Boolean expression relating these Boolean statements.

WAIT

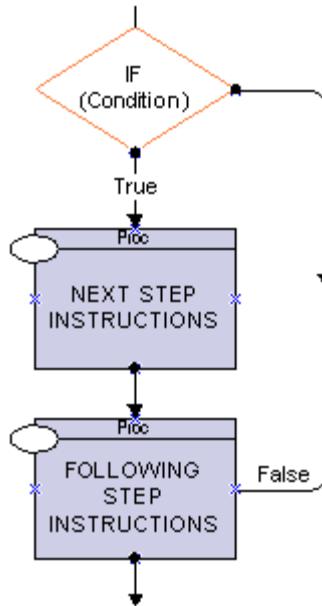
The statements of the Wait step, which may be at any level, define a condition to be checked at a specified polling frequency in order to proceed to the next step. A timeout period can be associated. Should the condition not be satisfied within the timeout period, it is possible to point to a different path of instructions than that followed in the case of a successful check. By default the timeout path is the same as the one after a successful check. Schematically:



The Wait step must contain statements of type TLM or CHECK_VARIABLE which define the condition check via a Boolean expression relating these Boolean statements.

PRECONDITION

The Precondition step is only available as a low-level step (2 or greater). It defines a condition to be checked, and the next step (which must be at the same level) is executed only if this condition is true. The following steps are then executed as normal. Schematically:



The Precondition step must contain statements of type TLM or CHECK_VARIABLE which define the condition check via a Boolean expression relating these Boolean statements.

4.1.2 VARIABLES

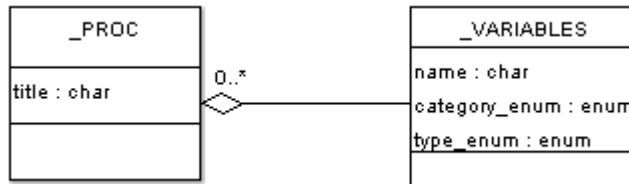


Figure 3 Variables in the Schema

Variables (VARIABLES element) can be local, global or procedure parameters (category_enum element). They can be referenced by:

- Any element in the procedure with a radix = VARIABLE
- The absolute_time_variable and delta_time_variable elements of the TimeType complex type.

Whenever a variable is derived from a command or telemetry parameter it inherits all that parameter's attributes. These attributes include exact size, data type and associated calibration, alias and range data sets defined for the parameter in the spacecraft database. This inherited information is passed through to the calling procedure if the variable is a procedure parameter.

Variables that are not derived from a command or a telemetry parameter must be assigned a basic data type (type_enum). As well as DERIVED this field can be BOOLEAN, INTEGER, REAL, DELTATIME, ABSOLUTETIME or STRING.

4.1.3 SET VARIABLE STATEMENT

The `set_variable_value` of the `_SET_VARIABLE` element can accept, in addition to a simple variable name, an expression containing standard arithmetic operators `+`, `-`, `*`, `()` and reference existing `_VARIABLE` elements by enclosing their names in square brackets. So for example if variables `var1`, `var2` and `var3` were defined, `set_variable_value` could be “`([var1] + [var2]) / var3`”

Syntax and type checking would be the responsibility of the reading program.

4.1.4 COMMAND FLAGS

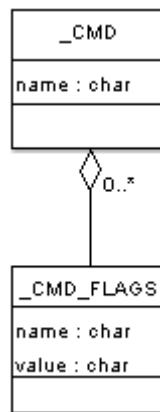


Figure 4 Command Flags in the Schema

Command flags are control system and mission specific. They are usually instructions to the control system to build the command in a particular way or to control its uplink and verification passage. Any number of different flags can be defined, as long as they are recognised by any mission specific software. Examples of command flags are: command blocking and command grouping, pre-transmission and verification stage requests, and critical and requires-authorisation attributes.

4.1.5 RADIX ENUMERATION

A radix defines the format for an entered value. In the named Schema enumeration radixEnum, entries matching RAW* mean that the value is uncalibrated. If the value is derived from a command parameter this refers to the uplinked value; if derived from a telemetry parameter it refers to the downlinked value. The enumerated values are:

- RAW - if numeric then this is the Decimal value
 - RAWHEX - hexadecimal value
 - RAWOCT - octal value
 - RAWBIN - binary value

- CALORALIAS - calibrated value as defined for the parameter in the spacecraft database. This can be a numeric calibration (e.g. point pair or polynomial) or an alias (enumeration).

The following are available for all except variable defaults (for which the restricted staticRadixEnum has been defined):

- TELEMETRYRAWVALUE - use the current raw value of the supplied telemetry parameter
- TELEMETRYCALVALUE - use the current calibrated value of the supplied telemetry parameter
- VARIABLE - a variable reference (see Variables section)

4.1.6 TIME TYPE

The complex type TimeType is defined for step and statement scheduling times as well as command uplink and execution times. It consists of a combination of an absolute time and a relative time. Both these elements can be a value or a variable. Additionally the absolute part can refer to a previously executing step or statement via a named label. At execution time this label is substituted with the actual execution time of the step or statement.

For procedures containing time-tagged commands (those destined for the on-board mission timeline), the absolute time element defaults to the time of the previously executed command in the procedure. The first time-tagged command of a procedure must therefore have an absolute time specified as a value or a variable, so that the following time-tags can be resolved.

For step and statement scheduled times the absolute time element defaults to the time at which the procedure starts execution.

4.1.7 DIRECTIVE CALL STATEMENT

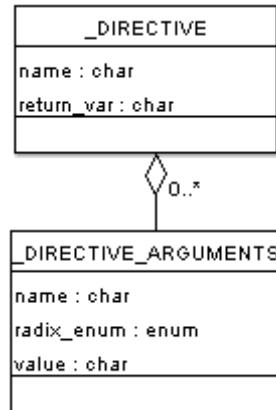


Figure 5 Directive call in the Schema

The `DIRECTIVE` statement is a call to an external Directive (with an optional return value `return_var`) whose call signature is defined separately.

5. DATA INTEGRITY CHECKS PERFORMED ON IMPORT

The procedure XML data are imported with a provided tool which generates MOIS procedures. During the import process the following checks are performed:

- The XML is structurally correct and conforms to the PROC schema
- The procedure is consistent with the currently loaded spacecraft TM/TC database. These checks are not fatal to the import process but the procedure will not be declared consistent until it can resolve all its spacecraft database references.
- All references to other procedures exist and can be resolved. This means that low level procedures called by other procedures must be imported first.

APPENDIX 1 SCHEMA PROC.XSD

Schema Proc.xsd

schema location: <Y:\ MOIS Documents\LLN SOS\Data\Common\Proc.xsd>
attribute form default: **unqualified**
element form default: **qualified**
targetNamespace: <http://www.omg.org/space/procspec>

Elements	Groups	Complex types	Simple types
Proc	AllStmts	ExecutionTimeType	binaryOperator
	ArithmeticComponent	ProcedureVariable	booleanOperator
	ArithmeticResult	SequenceVariable	radixEnum
	BooleanComponent	StepType	relationEnum
	BooleanResult	TimeType	staticRadixEnum
	BooleanStmts	Variable	unaryOperator
	CheckVarValue		valueRadixEnum
	Parameter		variableTypeEnum
	ProcHeader		varRadixEnum
	SeqHeader		
	SetVarValue		
	StepHeader		
	StmtHeader		

element Proc

diagram	<pre> classDiagram Proc < -- proc:Proc Proc --> proc:Variable : 0..oo Proc --> proc:Sequence : 0..oo Proc --> proc:Spacecraft : 0..oo Proc --> proc:ProcBody : 0..oo proc:Proc { attributes generated Generation date } proc:ProcBody { constraints unique VariableConstraint1 selector proc:Variable field proc:variableName } </pre> <p>Proc</p> <p>A procedure consists of Steps (procedure building blocks) which contain executable Statements. Certain elements within Statements can reference Variables.</p> <p>generated Generation date</p> <p>proc:Proc 0..oo Sub-programs expanded in the export</p> <p>proc:Variable 0..oo Procedure variables.</p> <p>proc:Sequence 0..oo Sequence Information</p> <p>proc:Spacecraft 0..oo Spacecraft (databases) from which the procedure statements are derived and with which they should remain consistent (multi-spacecraft missions only).</p> <p>proc:ProcBody 0..oo Procedure body.</p> <p>constraints</p> <ul style="list-style-type: none"> unique VariableConstraint1 selector proc:Variable field proc:variableName 												
namespace	http://www.omg.org/space/procspec												
properties	content complex												
children	proc:Proc proc:Variable proc:Sequence proc:Spacecraft proc:ProcBody												
used by	element Proc												
attributes	<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>annotation documentation</th> </tr> </thead> <tbody> <tr> <td>generated</td> <td></td> <td></td> <td></td> <td></td> <td>Generation date</td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	annotation documentation	generated					Generation date
Name	Type	Use	Default	Fixed	annotation documentation								
generated					Generation date								
identity constraints	<table border="1"> <thead> <tr> <th>Name</th> <th>Refer</th> <th>Selector</th> <th>Field(s)</th> </tr> </thead> <tbody> <tr> <td>unique</td> <td>VariableConstraint1</td> <td>proc:Variable</td> <td>proc:variableName</td> </tr> </tbody> </table>	Name	Refer	Selector	Field(s)	unique	VariableConstraint1	proc:Variable	proc:variableName				
Name	Refer	Selector	Field(s)										
unique	VariableConstraint1	proc:Variable	proc:variableName										
annotation	<p>documentation</p> <p>A procedure consists of Steps (procedure building blocks) which contain executable Statements. Certain elements within Statements can reference Variables.</p>												

source	<pre> <xs:element name="Proc"> <xs:annotation> <xs:documentation>A procedure consists of Steps (procdeure building blocks) which contain executable Statements. Certain elements within Statements can reference Variables. </xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element ref="proc:Proc" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Sub-programs expanded in the export</xs:documentation> </xs:annotation> </xs:element> <xs:choice> <xs:element name="Variable" type="proc:ProcedureVariable" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Procedure variables.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="Sequence" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Sequence Information</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="SeqHeader"> <xs:complexType> <xs:group ref="proc:SeqHeader"/> </xs:complexType> </xs:element> <xs:element name="seqName" type="xs:string"> <xs:annotation> <xs:documentation>Mapped to the seqName field in the step headers.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="SeqVariable" type="proc:SequenceVariable" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Sequence Variables</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:choice> <xs:element name="Spacecraft" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Spacecraft (databases) from which the procedure statements are derived and with which they should remain consistent (multi-spacecraft missions only).</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> </pre>
--------	--

```

<xs:element name="spacecraftName" type="xs:string"/>
<xs:element name="spacecraftDescription" type="xs:string"/>
<xs:element name="databaseVersion">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="name" type="xs:string">
        <xs:annotation>
          <xs:documentation>The name of the database version.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="comment" type="xs:string" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Comment associated to the database
version.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="domainId" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Domain Identifier associated to the
database.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="release" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Database Release</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="issue" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Database Issue</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="date" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Generation Date</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="version" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Database version</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ProcBody">
  <xs:annotation>
    <xs:documentation>Procedure body.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:group ref="proc:ProcHeader">
        <xs:annotation>

```

	<pre> <xs:documentation>Procedure header.</xs:documentation> </xs:annotation> </xs:group> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Procedure step. Steps are the procedure building blocks.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> <xs:complexType> <xs:element> </xs:sequence> <xs:attribute name="generated"> <xs:annotation> <xs:documentation>Generation date</xs:documentation> </xs:annotation> </xs:attribute> </xs:complexType> <xs:unique name="VariableConstraint1"> <xs:selector xpath="proc:Variable"/> <xs:field xpath="proc:variableName"/> </xs:unique> </xs:element> </pre>
--	--

attribute Proc/@generated

properties	isRef 0
annotation	documentation Generation date
source	<pre> <xs:attribute name="generated"> <xs:annotation> <xs:documentation>Generation date</xs:documentation> </xs:annotation> </xs:attribute> </pre>

element Proc/Variable

diagram	<pre> classDiagram class proc:ProcedureVariable { proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut proc:variableArrayUpperBound } class proc:Variable { *proc:ProcedureVariable } proc:Variable "0..∞" -- "*" proc:ProcedureVariable </pre> <p>The diagram illustrates the UML class <code>proc:ProcedureVariable</code>. This class contains several attributes:</p> <ul style="list-style-type: none"> <code>proc:variableName</code>: The name of the variable. <code>proc:variableCategoryEnum</code>: Variables are LOCAL to the procedure, GLOBAL to all procedures or PROCEDURE PARAMETERS (arguments). There may also be a RETURN VALUE. <code>proc:variableTypeEnum</code>: If the variable is DERIVED then its exact type is inherited from a TC or TM parameter - in this case the type is given in the <code>variableDerivedTypeEnum</code> field. Otherwise it takes a standard variable type (Integer, String etc.). For languages that do not declare their variables (such as STOL) the type is arbitrary; no type checking will be performed. <code>proc:variableDerivedTypeEnum</code>: The standard variable type (Integer, String etc.) in case of a DERIVED variable. <code>proc:variableProcParamOrder</code>: Defines an order for the parameter declarations (counted from 1) if required. <code>proc:variableDefaultRadixEnum</code> <code>proc:variableDefault</code> <code>proc:variableDescription</code> <code>proc:variableReadOnly</code>: True = constant. <code>proc:variableInOut</code>: True = in/out parameter, default = in parameter. <code>proc:variableArrayUpperBound</code>: If specified declares variable as an Array by specifying each of its dimensions. Not supported for DERIVED (VariableTypeEnum) types. <p>Relationships:</p> <ul style="list-style-type: none"> A multiplicity of <code>0..∞</code> for <code>proc:Variable</code> connects to a multiplicity of <code>*</code> for <code>proc:ProcedureVariable</code>.
namespace	http://www.omg.org/space/procspec

type	proc:ProcedureVariable								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut proc:variableArrayUpperBound								
annotation	documentation Procedure variables.								
source	<pre><xs:element name="Variable" type="proc:ProcedureVariable" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Procedure variables.</xs:documentation> </xs:annotation> </xs:element></pre>								

element Proc/Sequence

diagram	<p>Mapped to the seqName field in the step headers.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc:SeqHeader proc:seqName proc:SeqVariable								
annotation	documentation Sequence Information								
source	<pre><xs:element name="Sequence" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Sequence Information</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="SeqHeader"> <xs:complexType> <xs:group ref="proc:SeqHeader"/> </xs:complexType> </xs:element> <xs:element name="seqName" type="xs:string"> <xs:annotation> <xs:documentation>Mapped to the seqName field in the step headers.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="SeqVariable" type="proc:SequenceVariable" minOccurs="0"</pre>								

	<pre>maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Sequence Variables</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>
--	--

element Proc/Sequence/SeqHeader

diagram	<p>Sequence header.</p>
	<p>proc:seqDescr</p> <ul style="list-style-type: none"> proc:seqDetailedDescr proc:seqFlagCritical Mission-specific flag indicating a 'critical' sequence proc:seqFlagPlannable True if the sequence is plannable in a mission planning system. proc:seqStandAlone proc:seqSubSys proc:seqSubSchedId
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:seqDescr proc:seqDetailedDescr proc:seqFlagCritical proc:seqFlagPlannable proc:seqStandAlone proc:seqSubSys proc:seqSubSchedId
source	<pre><xs:element name="SeqHeader"> <xs:complexType> <xs:group ref="proc:SeqHeader"/> </xs:complexType> </xs:element></pre>

element Proc/Sequence/seqName

diagram	<p>Mapped to the seqName field in the step headers.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Mapped to the seqName field in the step headers.

source	<pre><xs:element name="seqName" type="xs:string"> <xs:annotation> <xs:documentation>Mapped to the seqName field in the step headers.</xs:documentation> </xs:annotation> </xs:element></pre>
--------	--

element Proc/Sequence/SeqVariable

diagram	<pre> classDiagram class proc:SequenceVariable { proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut proc:rangeSetOverride } class proc:SeqVariable { <> proc:SequenceVariable } proc:SeqVariable "0..∞" --> proc:SequenceVariable </pre>
namespace	http://www.omg.org/space/procspec
type	proc:SequenceVariable

properties	isRef 0 minOcc 0 maxOcc unbounded content complex
children	proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut proc:rangeSetOverride
annotation	documentation Sequence Variables
source	<xs:element name="SeqVariable" type="proc:SequenceVariable" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Sequence Variables</xs:documentation> </xs:annotation> </xs:element>

element Proc/Spacecraft

diagram	<pre> classDiagram proc:Spacecraft "0..∞" --> proc:spacecraftName proc:Spacecraft "0..∞" --> proc:spacecraftDescription proc:Spacecraft "0..∞" --> proc:databaseVersion </pre> <p>Spacecraft (databases) from which the procedure statements are derived and with which they should remain consistent (multi-spacecraft missions only).</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc unbounded content complex
children	proc:spacecraftName proc:spacecraftDescription proc:databaseVersion
annotation	documentation Spacecraft (databases) from which the procedure statements are derived and with which they should remain consistent (multi-spacecraft missions only).
source	<xs:element name="Spacecraft" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Spacecraft (databases) from which the procedure statements are derived and with which they should remain consistent (multi-spacecraft missions only).</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="spacecraftName" type="xs:string"/> <xs:element name="spacecraftDescription" type="xs:string"/> <xs:element name="databaseVersion"> <xs:complexType> <xs:sequence> <xs:element name="name" type="xs:string"> <xs:annotation> <xs:documentation>The name of the database version.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element>

```

<xs:element name="comment" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Comment associated to the database version.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="domainId" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Domain Identifier associated to the database.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="release" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Database Release</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="issue" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Database Issue</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="date" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Generation Date</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="version" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Database version</xs:documentation>
  </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>

```

element Proc/Spacecraft/spacecraftName

diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
source	<xs:element name="spacecraftName" type="xs:string"/>

element Proc/Spacecraft/spacecraftDescription

diagram	
---------	---

namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
source	<xs:element name="spacecraftDescription" type="xs:string"/>

element Proc/Spacecraft/databaseVersion

diagram	<pre> classDiagram class proc:databaseVersion { proc:name proc:comment proc:domainId proc:release proc:issue proc:date proc:version } proc:databaseVersion < --> proc:comment proc:databaseVersion < --> proc:domainId proc:databaseVersion < --> proc:release proc:databaseVersion < --> proc:issue proc:databaseVersion < --> proc:date proc:databaseVersion < --> proc:version </pre>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:name proc:comment proc:domainId proc:release proc:issue proc:date proc:version
source	<pre> <xs:element name="databaseVersion"> <xs:complexType> <xs:sequence> <xs:element name="name" type="xs:string"> <xs:annotation> <xs:documentation>The name of the database version.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="comment" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Comment associated to the database version.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="domainId" minOccurs="0"> <xs:annotation> <xs:documentation>Domain Identifier associated to the database.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

	<pre> </xs:annotation> </xs:element> <xs:element name="release" minOccurs="0"> <xs:annotation> <xs:documentation>Database Release</xs:documentation> </xs:annotation> </xs:element> <xs:element name="issue" minOccurs="0"> <xs:annotation> <xs:documentation>Database Issue</xs:documentation> </xs:annotation> </xs:element> <xs:element name="date" minOccurs="0"> <xs:annotation> <xs:documentation>Generation Date</xs:documentation> </xs:annotation> </xs:element> <xs:element name="version" minOccurs="0"> <xs:annotation> <xs:documentation>Database version</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

element Proc/Spacecraft/databaseVersion/name

diagram	<p>The name of the database version.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	<p>documentation</p> <p>The name of the database version.</p>
source	<pre> <xs:element name="name" type="xs:string"> <xs:annotation> <xs:documentation>The name of the database version.</xs:documentation> </xs:annotation> </xs:element> </pre>

element Proc/Spacecraft/databaseVersion/comment

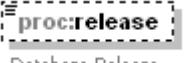
diagram	<p>Comment associated to the database version.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string

properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Comment associated to the database version.
source	<pre><xs:element name="comment" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Comment associated to the database version.</xs:documentation> </xs:annotation> </xs:element></pre>

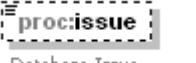
element Proc/Spacecraft/databaseVersion/domainId

diagram	 proc:domainId Domain Identifier associated to the database.
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc 1
annotation	documentation Domain Identifier associated to the database.
source	<pre><xs:element name="domainId" minOccurs="0"> <xs:annotation> <xs:documentation>Domain Identifier associated to the database.</xs:documentation> </xs:annotation> </xs:element></pre>

element Proc/Spacecraft/databaseVersion/release

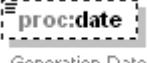
diagram	 proc:release Database Release
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc 1
annotation	documentation Database Release
source	<pre><xs:element name="release" minOccurs="0"> <xs:annotation> <xs:documentation>Database Release</xs:documentation> </xs:annotation> </xs:element></pre>

element Proc/Spacecraft/databaseVersion/issue

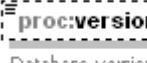
diagram	 proc:issue Database Issue
---------	--

namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc 1
annotation	documentation Database Issue
source	<pre><xs:element name="issue" minOccurs="0"> <xs:annotation> <xs:documentation>Database Issue</xs:documentation> </xs:annotation> </xs:element></pre>

element Proc/Spacecraft/databaseVersion/date

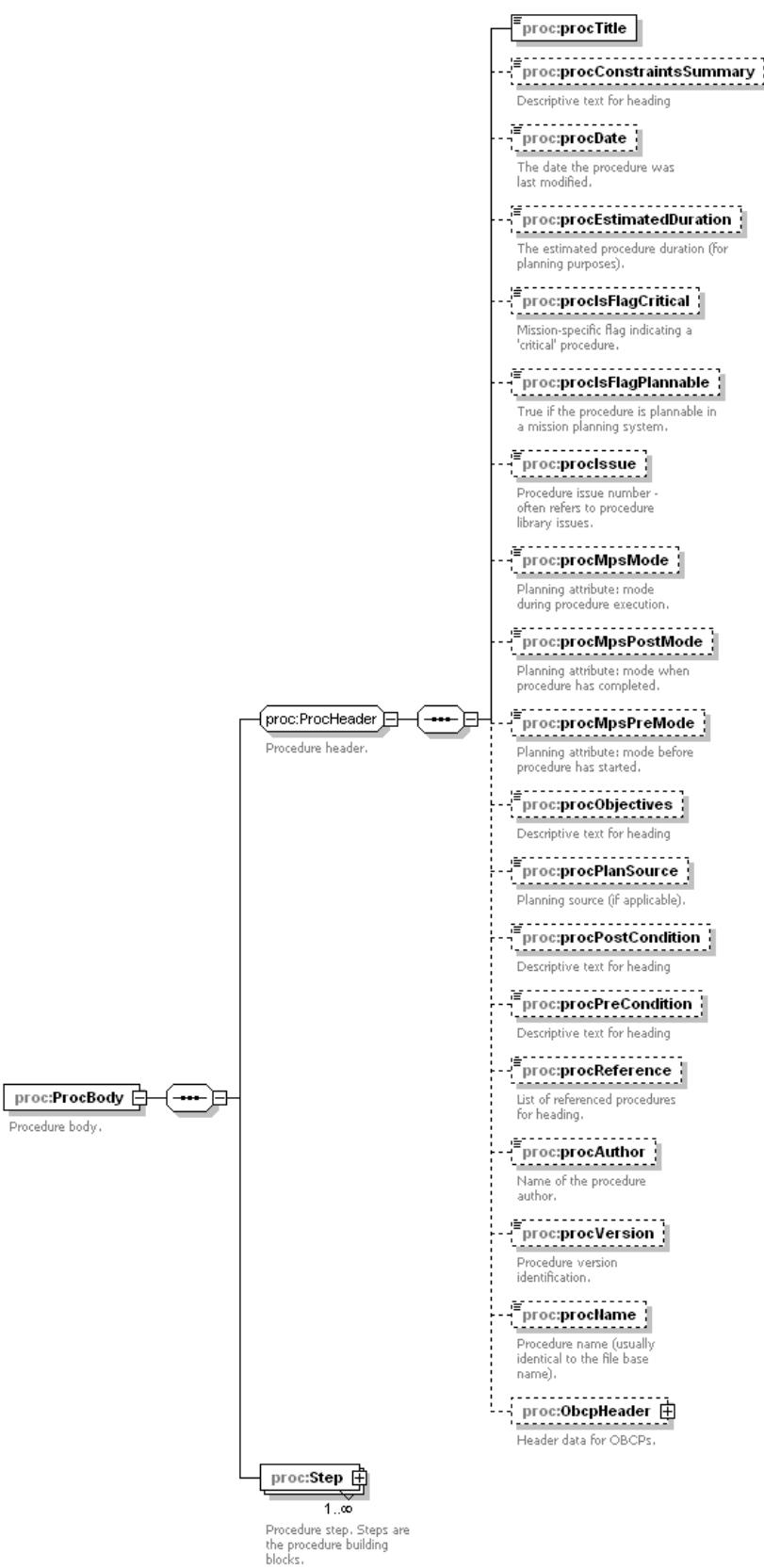
diagram	 proc:date Generation Date
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc 1
annotation	documentation Generation Date
source	<pre><xs:element name="date" minOccurs="0"> <xs:annotation> <xs:documentation>Generation Date</xs:documentation> </xs:annotation> </xs:element></pre>

element Proc/Spacecraft/databaseVersion/version

diagram	 proc:version Database version
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc 1
annotation	documentation Database version
source	<pre><xs:element name="version" minOccurs="0"> <xs:annotation> <xs:documentation>Database version</xs:documentation> </xs:annotation> </xs:element></pre>

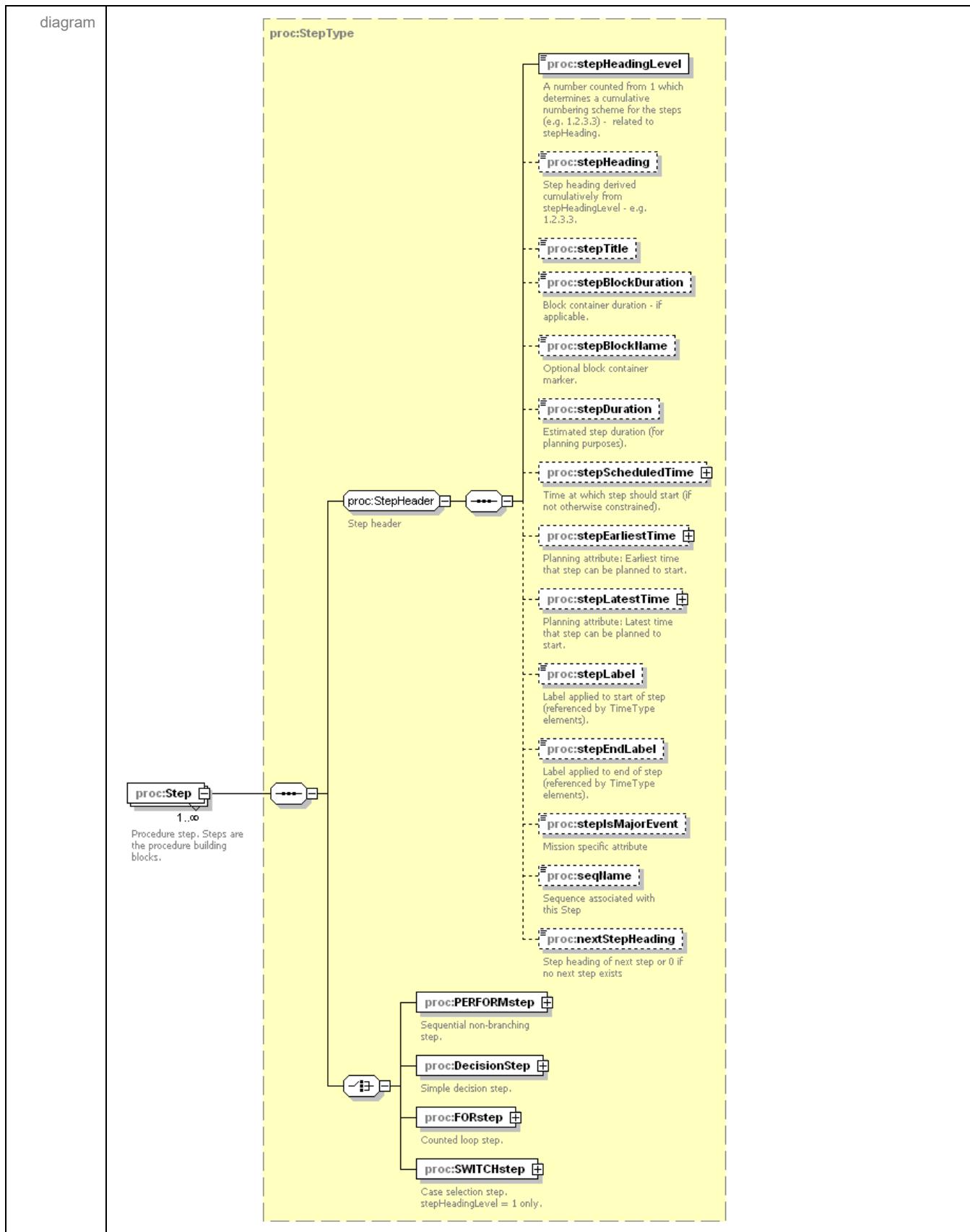
element Proc/ProcBody

diagram



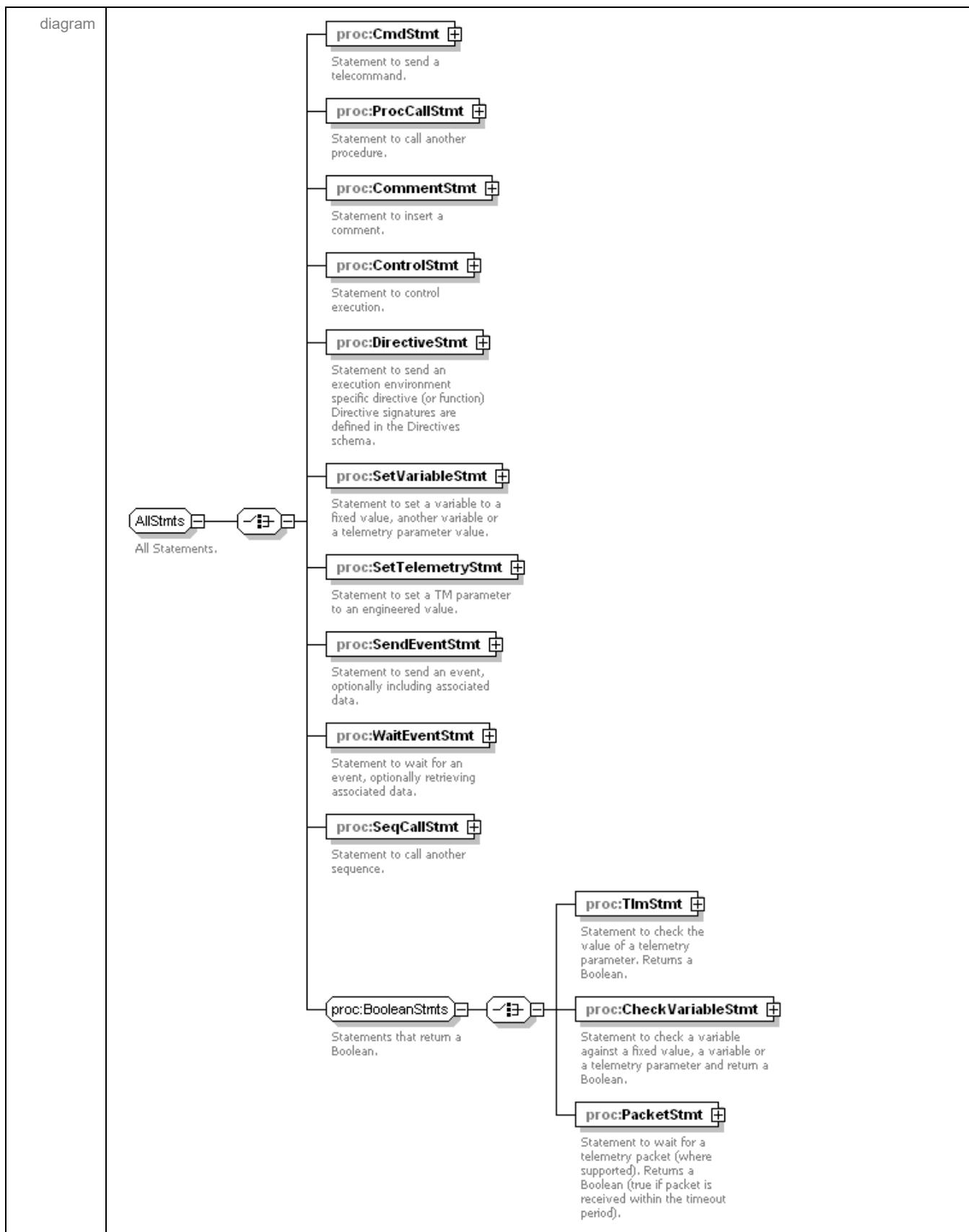
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:procTitle proc:procConstraintsSummary proc:procDate proc:procEstimatedDuration proc:procIsFlagCritical proc:procIsFlagPlannable proc:procIssue proc:procMpsMode proc:procMpsPostMode proc:procMpsPreMode proc:procObjectives proc:procPlanSource proc:procPostCondition proc:procPreCondition proc:procReference proc:procAuthor proc:procVersion proc:procName proc:ObcpHeader proc:Step
annotation	documentation Procedure body.
source	<pre><xs:element name="ProcBody"> <xs:annotation> <xs:documentation>Procedure body.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:ProcHeader"> <xs:annotation> <xs:documentation>Procedure header.</xs:documentation> </xs:annotation> </xs:group> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Procedure step. Steps are the procedure building blocks.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element **Proc/ProcBody/Step**



namespace	http://www.omg.org/space/procspec								
type	proc:StepType								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>1</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	1	maxOcc	unbounded	content	complex
isRef	0								
minOcc	1								
maxOcc	unbounded								
content	complex								
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep								
annotation	<p>documentation Procedure step. Steps are the procedure building blocks.</p>								
source	<pre><xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Procedure step. Steps are the procedure building blocks.</xs:documentation> </xs:annotation> </xs:element></pre>								

group **AllStmts**



namespace	http://www.omg.org/space/procspec
children	proc:CmdStmt proc:ProcCallStmt proc:CommentStmt proc:ControlStmt proc:DirectiveStmt proc:SetVariableStmt proc:SetTelemetryStmt proc:SendEventStmt proc:WaitEventStmt proc:SeqCallStmt proc:TlmStmt proc:CheckVariableStmt proc:PacketStmt
used by	element StepType/PERFORMstep
annotation	documentation All Statements.
source	<pre> <xs:group name="AllStmts"> <xs:annotation> <xs:documentation>All Statements.</xs:documentation> </xs:annotation> <xs:choice> <xs:element name="CmdStmt"> <xs:annotation> <xs:documentation>Statement to send a telecommand.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="cmdName" type="xs:string"> <xs:annotation> <xs:documentation>The command name</xs:documentation> </xs:annotation> </xs:element> <xs:element name="cmdDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The command description</xs:documentation> </xs:annotation> </xs:element> <xs:element name="cmdDetailedDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The command detailed description</xs:documentation> </xs:annotation> </xs:element> <xs:element name="CmdExecutionTime" type="proc:ExecutionTimeType" minOccurs="0"> <xs:annotation> <xs:documentation>On-board execution time (if destined for the on-board time-tagged queue (Master Schedule).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="SubScheduleId" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Value of the SubScheduleId of the command.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="CmdUplinkTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Uplink time separation from the last command (timimg managed by the control system)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="cmdParamSet" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of selected command parameter set (not supported by all control systems)</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:choice> </xs:group></pre>

```

</xs:annotation>
</xs:element>
<xs:element name="cmdParamValueSet" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Name of selected command parameter value set (not supported by all control systems)</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="cmdExecTimeout" type="xs:long" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Duration in seconds to wait for TC execution acknowledgement.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="CmdFlag" minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation>Set of mission specific command attributes (flags) indicating specific processing to be performed on the command by the control system . Examples are interlocks and blocks. </xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="cmdFlagName" type="xs:string"/>
      <xs:element name="cmdFlagValue" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="CmdParam" minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation>Command parameters.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:group ref="proc:Parameter"/>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ProcCallStmt">
  <xs:annotation>
    <xs:documentation>Statement to call another procedure.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:group ref="proc:StmtHeader"/>
      <xs:element name="procCallName" type="xs:string">
        <xs:annotation>
          <xs:documentation>Name of the called procedure.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="procCallDescr" type="xs:string" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Description of the called procedure.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:element name="procCallIsAsynchronous" type="xs:boolean" default="false">
    
```

```

minOccurs="0">
  <xs:annotation>
    <xs:documentation>If true then do not wait for the called procedure to
complete.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="procCallReturnVar" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The Variable into which the return value is written. If a Variable array
then the array index is enclosed by one or more brackets () directly after the
variable.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="ProcParam" minOccurs="0" maxOccurs="unbounded">
  <xs:complexType>
    <xs:sequence>
      <xs:group ref="proc:Parameter"/>
      <xs:element name="passByReference" type="xs:boolean" default="false"
minOccurs="0">
        <xs:annotation>
          <xs:documentation>Pass the parameter by reference so that it can be changed by the
called procedure. If applicable.</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="CommentStmt">
  <xs:annotation>
    <xs:documentation>Statement to insert a comment.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:group ref="proc:StmtHeader"/>
      <xs:element name="commentText" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ControlStmt">
  <xs:annotation>
    <xs:documentation>Statement to control execution.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:group ref="proc:StmtHeader"/>
      <xs:element name="controlComment" type="xs:string" minOccurs="0"/>
    <xs:choice>
      <xs:element name="ControlSync">
        <xs:annotation>
          <xs:documentation>Wait for outstanding requested notifications from the control
system).</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:choice>
  </xs:sequence>

```

```

<xs:sequence>
  <xs:element name="controlSyncType">
    <xs:annotation>
      <xs:documentation>Synchronize on command UPLINK or command EXECUTION notifications.</xs:documentation>
    </xs:annotation>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="UPLINK"/>
        <xs:enumeration value="EXECUTION"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="controlSyncMinWait" type="xs:duration" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Minimum wait before accepting Synchronize notifications</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="controlSyncTimeout" type="xs:duration" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Timeout for Synchronize notifications.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="controlSyncStopOnTimeout" type="xs:boolean" minOccurs="0">
    <xs:annotation>
      <xs:documentation>If true procedure is halted on timeout.</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ControlPause">
  <xs:annotation>
    <xs:documentation>Pause the procedure for a given duration.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="controlPauseDuration" type="xs:duration" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Pause duration</xs:documentation>
        </xs:annotation>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="ControlTerminate">
  <xs:annotation>
    <xs:documentation>Terminate the procedure</xs:documentation>
  </xs:annotation>
  <xs:complexType/>
</xs:element>
</xs:choice>
</xs:sequence>
</xs:complexType>
</xs:element>

```

	<pre> <xs:element name="DirectiveStmt"> <xs:annotation> <xs:documentation>Statement to send an execution environment specific directive (or function) Directive signatures are defined in the Directives schema.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="directiveName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called Directive.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="directiveIsAsynchronous" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If true then do not wait for the called Directive to complete.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="directiveReturnVar" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="DirectiveArgument" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="SetVariableStmt"> <xs:annotation> <xs:documentation>Statement to set a variable to a fixed value, another variable or a telemetry parameter value.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="setVariableName" type="xs:string"> <xs:annotation> <xs:documentation>The Variable to be set. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> <xs:group ref="proc:SetVarValue"> <xs:annotation> <xs:documentation>RHS of Set Variable</xs:documentation> </xs:annotation> </xs:group> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

```

</xs:complexType>
</xs:element>
<xs:element name="SetTelemetryStmt">
<xs:annotation>
<xs:documentation>Statement to set a TM parameter to an engineered
value.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:group ref="proc:StmtHeader"/>
<xs:element name="setTmParameterName" type="xs:string">
<xs:annotation>
<xs:documentation>The TM Parameter to be set.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="setTmValueRadixEnum" type="proc:radixEnum">
<xs:annotation>
<xs:documentation>Radix of set statement value </xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="setTmValue" type="xs:string">
<xs:annotation>
<xs:documentation>Fixed value on RHS (radix not Variable)</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="SendEventStmt">
<xs:annotation>
<xs:documentation>Statement to send an event, optionally including associated
data.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:group ref="proc:StmtHeader"/>
<xs:element name="eventName" type="xs:string">
<xs:annotation>
<xs:documentation>The event name (mnemonic)</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="arrayVarName" type="xs:string">
<xs:annotation>
<xs:documentation>Name of a variable of array type containing the event
data</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="numBytes" type="xs:string">
<xs:annotation>
<xs:documentation>Length in bytes of event data. May be a constant integer value or the
name of an integer variable.</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>

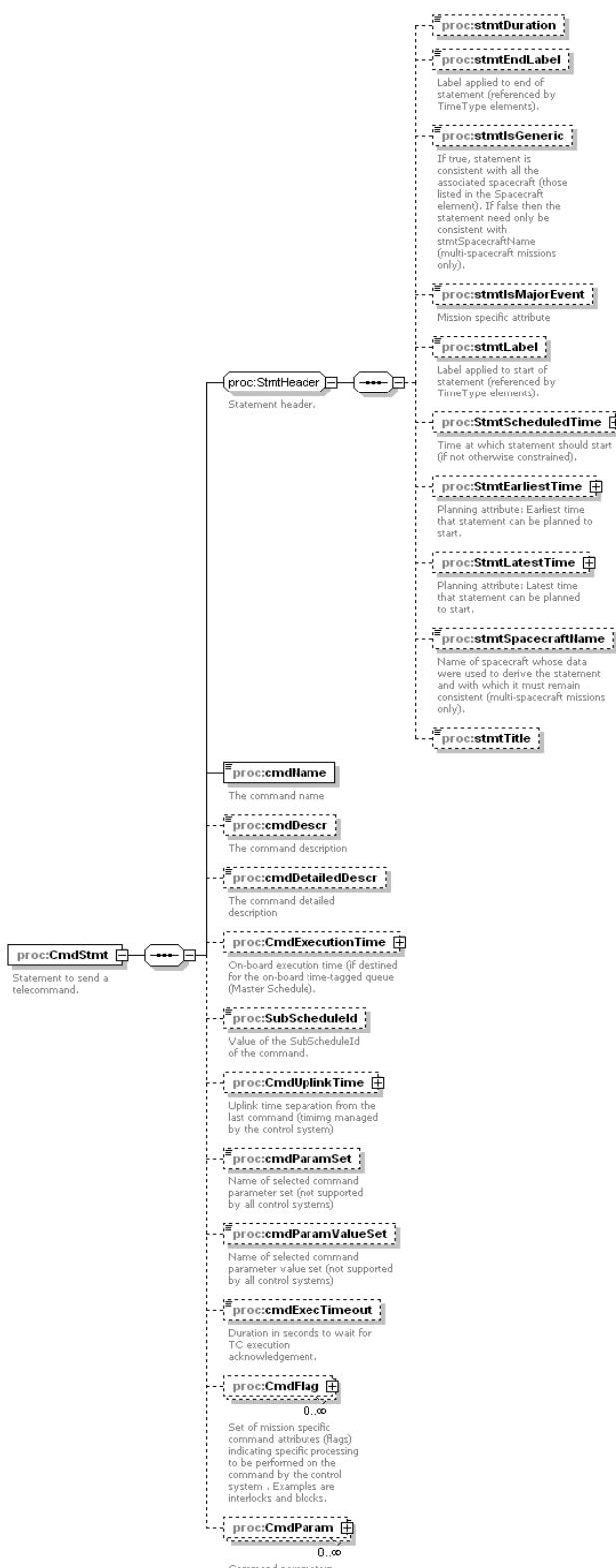
```

	<pre> <xs:element name="WaitEventStmt"> <xs:annotation> <xs:documentation>Statement to wait for an event, optionally retrieving associated data.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>The event name (mnemonic)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="arrayVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of a variable of array type to receive the event data</xs:documentation> </xs:annotation> </xs:element> <xs:element name="numBytesVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of an integer variable containing the number of bytes.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="timeout" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Optional timeout in seconds</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="SeqCallStmt"> <xs:annotation> <xs:documentation>Statement to call another sequence.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="seqCallName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called sequence.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="seqCallDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Description of the called sequence.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="SeqParam" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

	<pre></xs:complexType> </xs:element> <xs:group ref="proc:BooleanStmts"/> </xs:choice> </xs:group></pre>
--	---

element AllStmts/CmdStmt

diagram

namespace <http://www.omg.org/space/procspec>

properties	isRef 0 content complex
------------	----------------------------

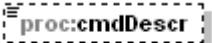
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:cmdName proc:cmdDescr proc:cmdDetailedDescr proc:CmdExecutionTime proc:SubScheduleId proc:CmdUplinkTime proc:cmdParamSet proc:cmdParamValueSet proc:cmdExecTimeout proc:CmdFlag proc:CmdParam
annotation	documentation Statement to send a telecommand.
source	<pre> <xs:element name="CmdStmt"> <xs:annotation> <xs:documentation>Statement to send a telecommand.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="cmdName" type="xs:string"> <xs:annotation> <xs:documentation>The command name</xs:documentation> </xs:annotation> </xs:element> <xs:element name="cmdDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The command description</xs:documentation> </xs:annotation> </xs:element> <xs:element name="cmdDetailedDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The command detailed description</xs:documentation> </xs:annotation> </xs:element> <xs:element name="CmdExecutionTime" type="proc:ExecutionTimeType" minOccurs="0"> <xs:annotation> <xs:documentation>On-board execution time (if destined for the on-board time-tagged queue (Master Schedule).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="SubScheduleId" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Value of the SubScheduleId of the command.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="CmdUplinkTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Uplink time separation from the last command (timimg managed by the control system)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="cmdParamSet" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of selected command parameter set (not supported by all control systems)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="cmdParamValueSet" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of selected command parameter value set (not supported by all control systems)</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

	<pre> </xs:annotation> </xs:element> <xs:element name="cmdExecTimeout" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Duration in seconds to wait for TC execution acknowledgement.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="CmdFlag" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Set of mission specific command attributes (flags) indicating specific processing to be performed on the command by the control system . Examples are interlocks and blocks. </xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="cmdFlagName" type="xs:string"/> <xs:element name="cmdFlagValue" type="xs:string"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="CmdParam" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Command parameters.</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

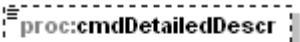
element AllStmts/CmdStmt/cmdName

diagram	<p>The command name</p>				
namespace	http://www.omg.org/space/procspec				
type	xs:string				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
annotation	<p>documentation</p> <p>The command name</p>				
source	<pre> <xs:element name="cmdName" type="xs:string"> <xs:annotation> <xs:documentation>The command name</xs:documentation> </xs:annotation> </xs:element> </pre>				

element **AllStmts/CmdStmt/cmdDescr**

diagram	 The command description
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The command description
source	<pre><xs:element name="cmdDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The command description</xs:documentation> </xs:annotation> </xs:element></pre>

element **AllStmts/CmdStmt/cmdDetailedDescr**

diagram	 The command detailed description
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The command detailed description
source	<pre><xs:element name="cmdDetailedDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The command detailed description</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/CmdStmt/CmdExecutionTime

diagram	<pre> classDiagram class proc:ExecutionTimeType { <<proc:CmdExecutionTime>> <<On-board execution time (if destined for the on-board time-tagged queue (Master Schedule).)>> <<proc:absoluteTime>> <<proc:absoluteTimeLabel>> <<proc:absoluteTimeVariable>> <<proc:negativeSign>> <<proc:relativeTime>> <<proc:relativeTimeVariable>> <<proc:orbitAngleVariable>> <<proc:orbitAngleRadix>> <<proc:negativeSign>> <<proc:orbitAngle>> } class proc:CmdExecutionTime class proc:absoluteTime class proc:absoluteTimeLabel class proc:absoluteTimeVariable class proc:negativeSign class proc:relativeTime class proc:relativeTimeVariable class proc:orbitAngleVariable class proc:orbitAngleRadix class proc:negativeSign class proc:orbitAngle </pre> <p>The diagram illustrates the structure of the <code>proc:ExecutionTimeType</code> element. It is defined by the <code>proc:CmdExecutionTime</code> class, which is annotated with the documentation: "On-board execution time (if destined for the on-board time-tagged queue (Master Schedule)).". The <code>proc:ExecutionTimeType</code> class itself contains several components:</p> <ul style="list-style-type: none"> <code>proc:absoluteTime</code>: An absolute time literal. <code>proc:absoluteTimeLabel</code>: A label defined in a Step or Statement. <code>proc:absoluteTimeVariable</code>: A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. <code>proc:negativeSign</code>: A negative sign indicator. <code>proc:relativeTime</code>: A relative time literal. <code>proc:relativeTimeVariable</code>: A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. <code>proc:orbitAngleVariable</code>: A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. <code>proc:orbitAngleRadix</code>: A radix for an orbit angle. <code>proc:negativeSign</code>: A negative sign indicator. <code>proc:orbitAngle</code>: An orbit angle value. Orbit number is allowed e.g. 1-20 for orbit 1 angle 20. 								
namespace	http://www.omg.org/space/procspec								
type	proc:ExecutionTimeType								
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable proc:orbitAngleVariable proc:orbitAngleRadix proc:negativeSign proc:orbitAngle								
annotation	documentation On-board execution time (if destined for the on-board time-tagged queue (Master Schedule)).								
source	<pre> <x:element name="CmdExecutionTime" type="proc:ExecutionTimeType" minOccurs="0"> <x:annotation> </pre>								

	<p><xs:documentation>On-board execution time (if destined for the on-board time-tagged queue (Master Schedule).</xs:documentation></p> <p></xs:annotation></p> <p></xs:element></p>
--	---

element AllStmts/CmdStmt/SubScheduleId

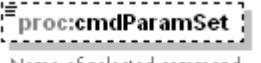
diagram									
namespace	http://www.omg.org/space/procspec								
type	xs:long								
properties	<table border="0"> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>Value of the SubScheduleId of the command.</p>								
source	<pre><xs:element name="SubScheduleId" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Value of the SubScheduleId of the command.</xs:documentation> </xs:annotation> </xs:element></pre>								

element AllStmts/CmdStmt/CmdUplinkTime

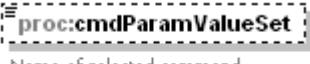
diagram	
---------	--

namespace	http://www.omg.org/space/procspec
type	proc:TimeType
properties	isRef 0 minOcc 0 maxOcc 1 content complex
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable
annotation	documentation Uplink time separation from the last command (timimg managed by the control system)
source	<pre><xs:element name="CmdUplinkTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Uplink time separation from the last command (timimg managed by the control system)</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/CmdStmt/cmdParamSet

diagram	 Name of selected command parameter set (not supported by all control systems)
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Name of selected command parameter set (not supported by all control systems)
source	<pre><xs:element name="cmdParamSet" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of selected command parameter set (not supported by all control systems)</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/CmdStmt/cmdParamValueSet

diagram	 Name of selected command parameter value set (not supported by all control systems)
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple

annotation	documentation Name of selected command parameter value set (not supported by all control systems)
source	<pre><xs:element name="cmdParamValueSet" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of selected command parameter value set (not supported by all control systems)</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/CmdStmt/cmdExecTimeout

diagram	<p>Duration in seconds to wait for TC execution acknowledgement.</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:long								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	documentation Duration in seconds to wait for TC execution acknowledgement.								
source	<pre><xs:element name="cmdExecTimeout" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Duration in seconds to wait for TC execution acknowledgement.</xs:documentation> </xs:annotation> </xs:element></pre>								

element AllStmts/CmdStmt/CmdFlag

diagram	<p>Set of mission specific command attributes (flags) indicating specific processing to be performed on the command by the control system . Examples are interlocks and blocks.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc:cmdFlagName proc:cmdFlagValue								
annotation	documentation Set of mission specific command attributes (flags) indicating specific processing to be performed on the command by the control system . Examples are interlocks and blocks.								
source	<pre><xs:element name="CmdFlag" minOccurs="0" maxOccurs="unbounded"> <xs:annotation></pre>								

	<p><xs:documentation>Set of mission specific command attributes (flags) indicating specific processing to be performed on the command by the control system . Examples are interlocks and blocks. </xs:documentation></p> <pre></xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="cmdFlagName" type="xs:string"/> <xs:element name="cmdFlagValue" type="xs:string"/> </xs:sequence> </xs:complexType> </xs:element></pre>
--	--

element AllStmts/CmdStmt/CmdFlag/cmdFlagName

diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
source	<xs:element name="cmdFlagName" type="xs:string"/>

element AllStmts/CmdStmt/CmdFlag/cmdFlagValue

diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
source	<xs:element name="cmdFlagValue" type="xs:string"/>

element AllStmts/CmdStmt/CmdParam

diagram	<pre> classDiagram class proc:CmdParam { <<Command parameters.>> <<0..>> } class proc:Parameter { <<Generic parameter.>> } class proc:ParamValueParam { <<Used for nested parameters, e.g. TC as TC parameter.>> <<0..>> } class proc:paramName class proc:paramDescr class proc:paramEngUnit class proc:paramValueRadixEnum class proc:paramValue class proc:paramRawValue class proc:groupRepeatSize proc:CmdParam < -- proc:Parameter proc:Parameter --> proc:ParamValueParam proc:ParamValueParam --> proc:groupRepeatSize </pre> <p>The diagram illustrates the structure of the <code>CmdParam</code> element. It inherits from <code>Parameter</code>, which in turn has a child <code>ParamValueParam</code>. This <code>ParamValueParam</code> node then points to a <code>groupRepeatSize</code> node. The <code>CmdParam</code> element is annotated with "Command parameters." and a multiplicity of "0..". The <code>Parameter</code> element is annotated with "Generic parameter." and a multiplicity of "0..". The <code>ParamValueParam</code> element is annotated with "Used for nested parameters, e.g. TC as TC parameter." and a multiplicity of "0..". The children of <code>CmdParam</code> are detailed in the following table.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table border="0"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc: paramName proc: paramDescr proc: paramEngUnit proc: paramValueRadixEnum proc: paramValue proc: paramRawValue proc: groupRepeatSize proc: ParamValueParam								
annotation	<p>documentation Command parameters.</p>								
source	<pre> <xs:element name="CmdParam" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Command parameters.</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType> </pre>								

	<code></xs:element></code>
element AllStmts/ProcCallStmt	
diagram	<pre> classDiagram class proc:StmtHeader { <<Statement header.>> } class proc:ProcCallStmt { <<Statement to call another procedure.>> } proc:StmtHeader < -- proc:ProcCallStmt proc:ProcCallStmt < -- proc:stmtDuration proc:ProcCallStmt < -- proc:stmtEndLabel proc:ProcCallStmt < -- proc:stmtIsGeneric proc:ProcCallStmt < -- proc:stmtIsMajorEvent proc:ProcCallStmt < -- proc:stmtLabel proc:ProcCallStmt < -- proc:stmtScheduledTime proc:ProcCallStmt < -- proc:stmtEarliestTime proc:ProcCallStmt < -- proc:stmtLatestTime proc:ProcCallStmt < -- proc:stmtSpacecraftName proc:ProcCallStmt < -- proc:stmtTitle proc:ProcCallStmt < -- proc:procCallName proc:ProcCallStmt < -- proc:procCallDescr proc:ProcCallStmt < -- proc:procCallsAsynchronous proc:ProcCallStmt < -- proc:procCallReturnVar proc:ProcCallStmt < -- proc:ProcParam </pre>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:procCallName proc:procCallDescr proc:procCallsAsynchronous proc:procCallReturnVar proc:ProcParam
annotation	documentation Statement to call another procedure.
source	<code><xs:element name="ProcCallStmt"></code>

	<pre> <xs:annotation> <xs:documentation>Statement to call another procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="procCallName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called procedure.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procCallDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Description of the called procedure.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procCallsAsynchronous" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If true then do not wait for the called procedure to complete.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procCallReturnVar" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="ProcParam" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:group ref="proc:Parameter"/> <xs:element name="passByReference" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>Pass the parameter by reference so that it can be changed by the called procedure. If applicable.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

element AllStmts/ProcCallStmt/procCallName

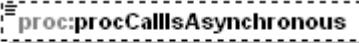
diagram	<p>proc:procCallName</p> <p>Name of the called procedure.</p>
namespace	http://www.omg.org/space/procspec

type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of the called procedure.
source	<pre><xs:element name="procCallName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called procedure.</xs:documentation> </xs:annotation> </xs:element></pre>

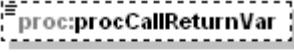
element AllStmts/ProcCallStmt/procCallDescr

diagram	 <p>Description of the called procedure.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Description of the called procedure.
source	<pre><xs:element name="procCallDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Description of the called procedure.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/ProcCallStmt/procCallsAsynchronous

diagram	 <p>If true then do not wait for the called procedure to complete.</p>
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple default false
annotation	documentation If true then do not wait for the called procedure to complete.
source	<pre><xs:element name="procCallsAsynchronous" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If true then do not wait for the called procedure to complete.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/ProcCallStmt/procCallReturnVar

diagram									
	The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</p>								
source	<pre><xs:element name="procCallReturnVar" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element></pre>								

element AllStmts/ProcCallStmt/ProcParam

diagram	<pre> classDiagram class proc:Parameter { <<Generic parameter.>> } class proc:ProcParam { <<0..>> } class proc:ParameterValueParam { <<0..>> } proc:Parameter < -- proc:ProcParam proc:ProcParam --> proc:ParameterValueParam : passByReference proc:ProcParam --> proc:ParameterValueParam : groupRepeatSize </pre> <p>The diagram illustrates the UML class structure for <code>proc:ProcParam</code>. It inherits from <code>proc:Parameter</code> (indicated by a solid line) and has two associations: one with <code>proc:ParameterValueParam</code> labeled <code>passByReference</code>, and another with <code>proc:ParameterValueParam</code> labeled <code>groupRepeatSize</code>.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table border="0"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc: paramName proc: paramDescr proc: paramEngUnit proc: paramValueRadixEnum proc: paramValue proc: paramRawValue proc: groupRepeatSize proc: ParamValueParam proc: passByReference								
source	<pre> <xs:element name="ProcParam" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:group ref="proc:Parameter"/> <xs:element name="passByReference" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> </pre>								

	<p><xs:documentation>Pass the parameter by reference so that it can be changed by the called procedure. If applicable.</xs:documentation></p> <pre></xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>
--	--

element AllStmts/ProcCallStmt/ProcParam/passByReference

diagram	 <p>Pass the parameter by reference so that it can be changed by the called procedure. If applicable.</p>										
namespace	http://www.omg.org/space/procspec										
type	xs:boolean										
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> <tr><td>default</td><td>false</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	false
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	false										
annotation	<p>documentation</p> <p>Pass the parameter by reference so that it can be changed by the called procedure. If applicable.</p>										
source	<pre><xs:element name="passByReference" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>Pass the parameter by reference so that it can be changed by the called procedure. If applicable.</xs:documentation> </xs:annotation> </xs:element></pre>										

element AllStmts/CommentStmt

diagram	<p>proc:CommentStmt</p> <p>Statement to insert a comment.</p> <p>proc:StmtHeader</p> <p>Statement header.</p> <p>proc:commentText</p> <p>proc:stmtDuration</p> <p>proc:stmtEndLabel</p> <p>proc:stmtIsGeneric</p> <p>proc:stmtIsMajorEvent</p> <p>proc:stmtLabel</p> <p>proc:stmtScheduledTime</p> <p>proc:stmtEarliestTime</p> <p>proc:stmtLatestTime</p> <p>proc:stmtSpacecraftName</p> <p>proc:stmtTitle</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:stmtScheduledTime proc:stmtEarliestTime proc:stmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:commentText				

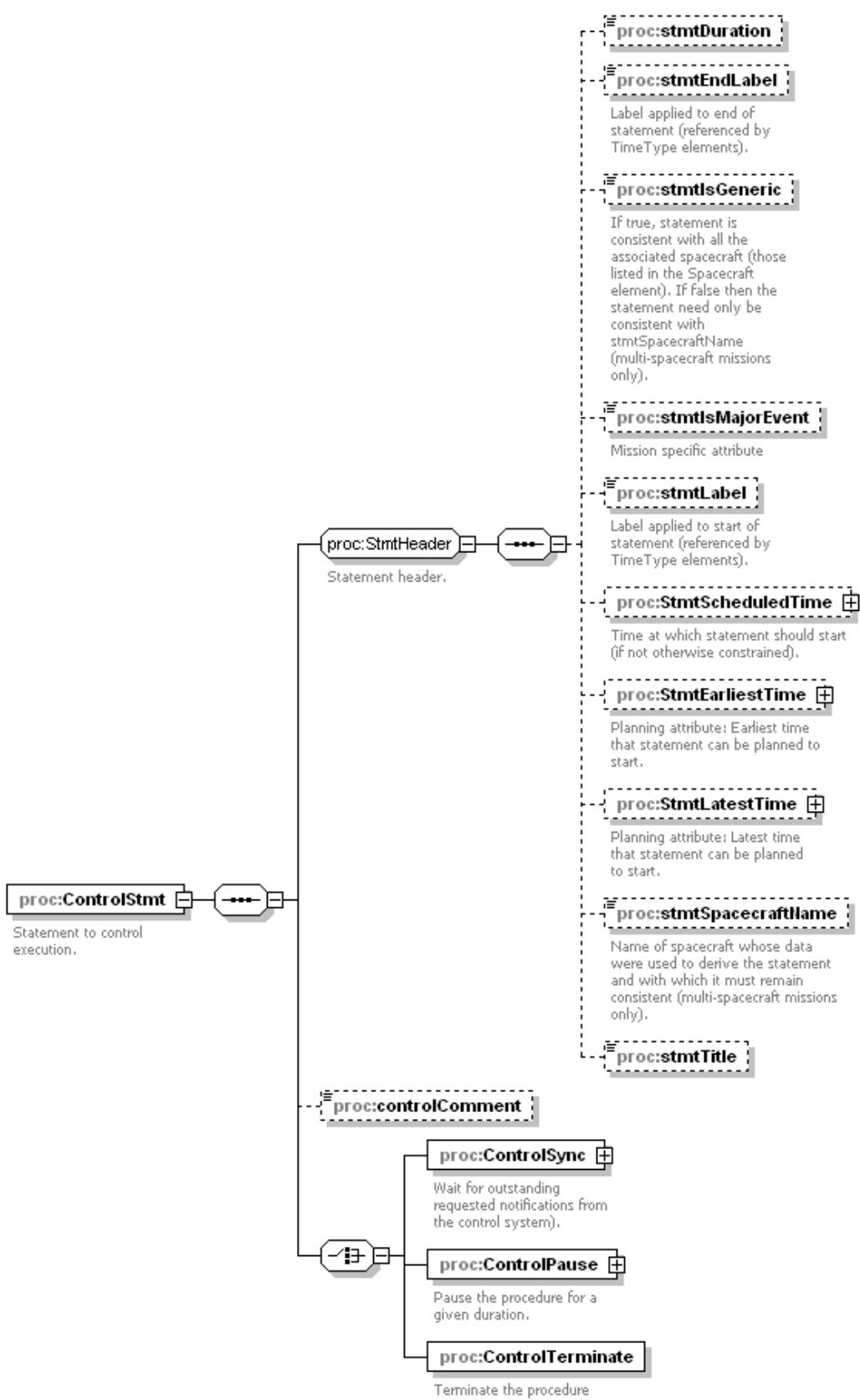
annotation	documentation Statement to insert a comment.
source	<pre><xs:element name="CommentStmt"> <xs:annotation> <xs:documentation>Statement to insert a comment.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="commentText" type="xs:string"/> </xs:sequence> </xs:complexType> </xs:element></pre>

element AllStmts/CommentStmt/commentText

diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
source	<xs:element name="commentText" type="xs:string"/>

element AllStmts/ControlStmt

diagram



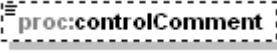
namespace	http://www.omg.org/space/procspec				
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:controlComment proc:ControlSync proc:ControlPause proc:ControlTerminate				
annotation	documentation Statement to control execution.				
source	<pre> <xs:element name="ControlStmt"> <xs:annotation> <xs:documentation>Statement to control execution.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="controlComment" type="xs:string" minOccurs="0"/> <xs:choice> <xs:element name="ControlSync"> <xs:annotation> <xs:documentation>Wait for outstanding requested notifications from the control system).</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="controlSyncType"> <xs:annotation> <xs:documentation>Synchronize on command UPLINK or command EXECUTION notifications.</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="UPLINK"/> <xs:enumeration value="EXECUTION"/> </xs:restriction> </xs:simpleType> </xs:element> <xs:element name="controlSyncMinWait" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Minimum wait before accepting Synchronize notifications</xs:documentation> </xs:annotation> </xs:element> <xs:element name="controlSyncTimeout" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout for Synchronize notifications.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="controlSyncStopOnTimeout" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>If true procedure is halted on timeout.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="ControlPause"> </xs:choice> </xs:sequence> </xs:complexType> </xs:element></pre>				

```

<xs:annotation>
  <xs:documentation>Pause the procedure for a given duration.</xs:documentation>
</xs:annotation>
<xs:complexType>
  <xs:sequence>
    <xs:element name="controlPauseDuration" type="xs:duration" minOccurs="0">
      <xs:annotation>
        <xs:documentation>Pause duration</xs:documentation>
      </xs:annotation>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ControlTerminate">
  <xs:annotation>
    <xs:documentation>Terminate the procedure</xs:documentation>
  </xs:annotation>
  <xs:complexType/>
</xs:element>
</xs:choice>
</xs:sequence>
</xs:complexType>
</xs:element>

```

element **AllStmts/ControlStmt/controlComment**

diagram									
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
source	<xs:element name="controlComment" type="xs:string" minOccurs="0"/>								

element AllStmts/ControlStmt/ControlSync

diagram	<p>proc:controlSync</p> <p>Wait for outstanding requested notifications from the control system).</p> <p>proc:controlSyncType Synchronize on command UPLINK or command EXECUTION notifications.</p> <p>proc:controlSyncMinWait Minimum wait before accepting Synchronize notifications</p> <p>proc:controlSyncTimeout Timeout for Synchronize notifications.</p> <p>proc:controlSyncStopOnTimeout If true procedure is halted on timeout.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:controlSyncType proc:controlSyncMinWait proc:controlSyncTimeout proc:controlSyncStopOnTimeout
annotation	<p>documentation</p> <p>Wait for outstanding requested notifications from the control system).</p>
source	<pre><xs:element name="ControlSync"> <xs:annotation> <xs:documentation>Wait for outstanding requested notifications from the control system).</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="controlSyncType"> <xs:annotation> <xs:documentation>Synchronize on command UPLINK or command EXECUTION notifications.</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="UPLINK"/> <xs:enumeration value="EXECUTION"/> </xs:restriction> <xs:simpleType> </xs:element> <xs:element name="controlSyncMinWait" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Minimum wait before accepting Synchronize notifications</xs:documentation> </xs:annotation> </xs:element> <xs:element name="controlSyncTimeout" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout for Synchronize notifications.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="controlSyncStopOnTimeout" type="xs:boolean" minOccurs="0"> <xs:annotation></pre>

	<pre><xs:documentation>If true procedure is halted on timeout.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>
--	---

element AllStmts/ControlStmt/ControlSync/controlSyncType

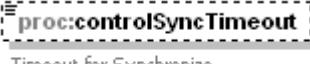
diagram	<p>proc:controlSyncType</p> <p>Synchronize on command UPLINK or command EXECUTION notifications.</p>				
namespace	http://www.omg.org/space/procspec				
type	restriction of xs:string				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
facets	<table> <tr> <td>enumeration</td> <td>UPLINK</td> </tr> <tr> <td>enumeration</td> <td>EXECUTION</td> </tr> </table>	enumeration	UPLINK	enumeration	EXECUTION
enumeration	UPLINK				
enumeration	EXECUTION				
annotation	<p>documentation</p> <p>Synchronize on command UPLINK or command EXECUTION notifications.</p>				
source	<pre><xs:element name="controlSyncType"> <xs:annotation> <xs:documentation>Synchronize on command UPLINK or command EXECUTION notifications.</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="UPLINK"/> <xs:enumeration value="EXECUTION"/> </xs:restriction> </xs:simpleType> </xs:element></pre>				

element AllStmts/ControlStmt/ControlSync/controlSyncMinWait

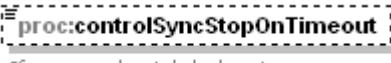
diagram	<p>proc:controlSyncMinWait</p> <p>Minimum wait before accepting Synchronize notifications</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:duration								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>Minimum wait before accepting Synchronize notifications</p>								
source	<pre><xs:element name="controlSyncMinWait" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Minimum wait before accepting Synchronize notifications</xs:documentation></pre>								

	<pre></xs:annotation> </xs:element></pre>
--	---

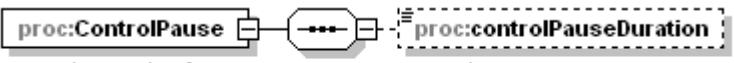
element AllStmts/ControlStmt/ControlSync/controlSyncTimeout

diagram	 <p>Timeout for Synchronize notifications.</p>
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Timeout for Synchronize notifications.
source	<pre><xs:element name="controlSyncTimeout" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout for Synchronize notifications.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/ControlStmt/ControlSync/controlSyncStopOnTimeout

diagram	 <p>If true procedure is halted on timeout.</p>
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation If true procedure is halted on timeout.
source	<pre><xs:element name="controlSyncStopOnTimeout" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>If true procedure is halted on timeout.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/ControlStmt/ControlPause

diagram	 <p>Pause the procedure for a given duration.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex

children	proc:controlPauseDuration
annotation	documentation Pause the procedure for a given duration.
source	<pre><xs:element name="ControlPause"> <xs:annotation> <xs:documentation>Pause the procedure for a given duration.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="controlPauseDuration" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Pause duration</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element AllStmts/ControlStmt/ControlPause/controlPauseDuration

diagram	 proc:controlPauseDuration Pause duration
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Pause duration
source	<pre><xs:element name="controlPauseDuration" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Pause duration</xs:documentation> </xs:annotation> </xs:element></pre>

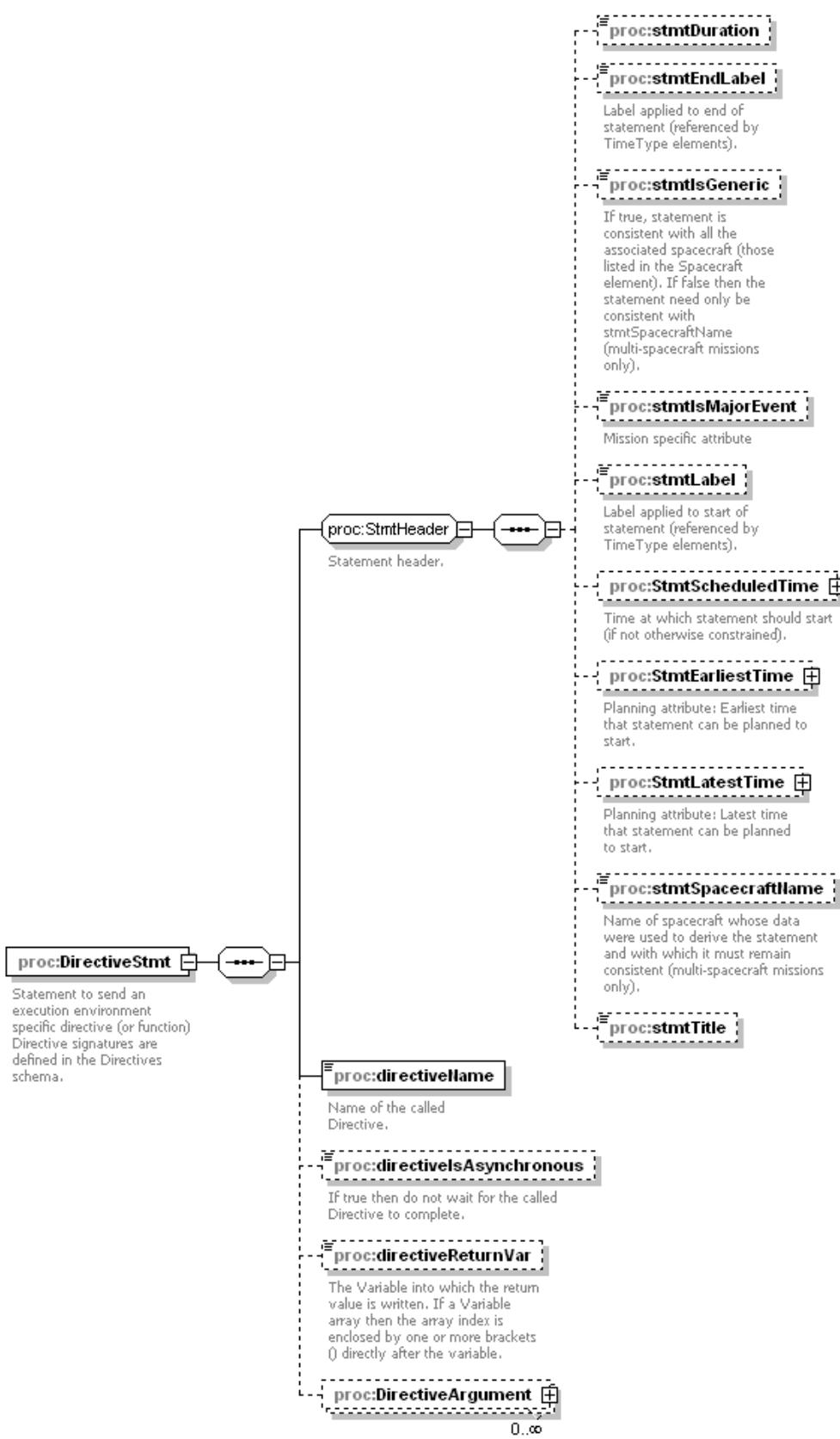
element AllStmts/ControlStmt/ControlTerminate

diagram	 proc:ControlTerminate Terminate the procedure
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
annotation	documentation Terminate the procedure
source	<pre><xs:element name="ControlTerminate"> <xs:annotation> <xs:documentation>Terminate the procedure</xs:documentation> </xs:annotation> <xs:complexType/></pre>

	</xs:element>
--	---------------

element AllStmts/DirectiveStmt

diagram



namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:directiveName proc:directivelsAsynchronous proc:directiveReturnVar proc:DirectiveArgument
annotation	documentation Statement to send an execution environment specific directive (or function) Directive signatures are defined in the Directives schema.
source	<pre><xs:element name="DirectiveStmt"> <xs:annotation> <xs:documentation>Statement to send an execution environment specific directive (or function) Directive signatures are defined in the Directives schema.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="directiveName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called Directive.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="directivelsAsynchronous" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If true then do not wait for the called Directive to complete.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="directiveReturnVar" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="DirectiveArgument" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:element></pre>

element AllStmts/DirectiveStmt/directiveName

diagram	 <p>Name of the called Directive.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string

properties	isRef 0 content simple
annotation	documentation Name of the called Directive.
source	<xs:element name="directiveName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called Directive.</xs:documentation> </xs:annotation> </xs:element>

element AllStmts/DirectiveStmt/directiveIsAsynchronous

diagram	 proc:directiveIsAsynchronous
	If true then do not wait for the called Directive to complete.
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple default false
annotation	documentation If true then do not wait for the called Directive to complete.
source	<xs:element name="directiveIsAsynchronous" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If true then do not wait for the called Directive to complete.</xs:documentation> </xs:annotation> </xs:element>

element AllStmts/DirectiveStmt/directiveReturnVar

diagram	 proc:directiveReturnVar
	The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<xs:element name="directiveReturnVar" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The Variable into which the return value is written. If a Variable array then the array index is enclosed by one or more brackets () directly after the

	variable.</xs:documentation> </xs:annotation> </xs:element>
--	---

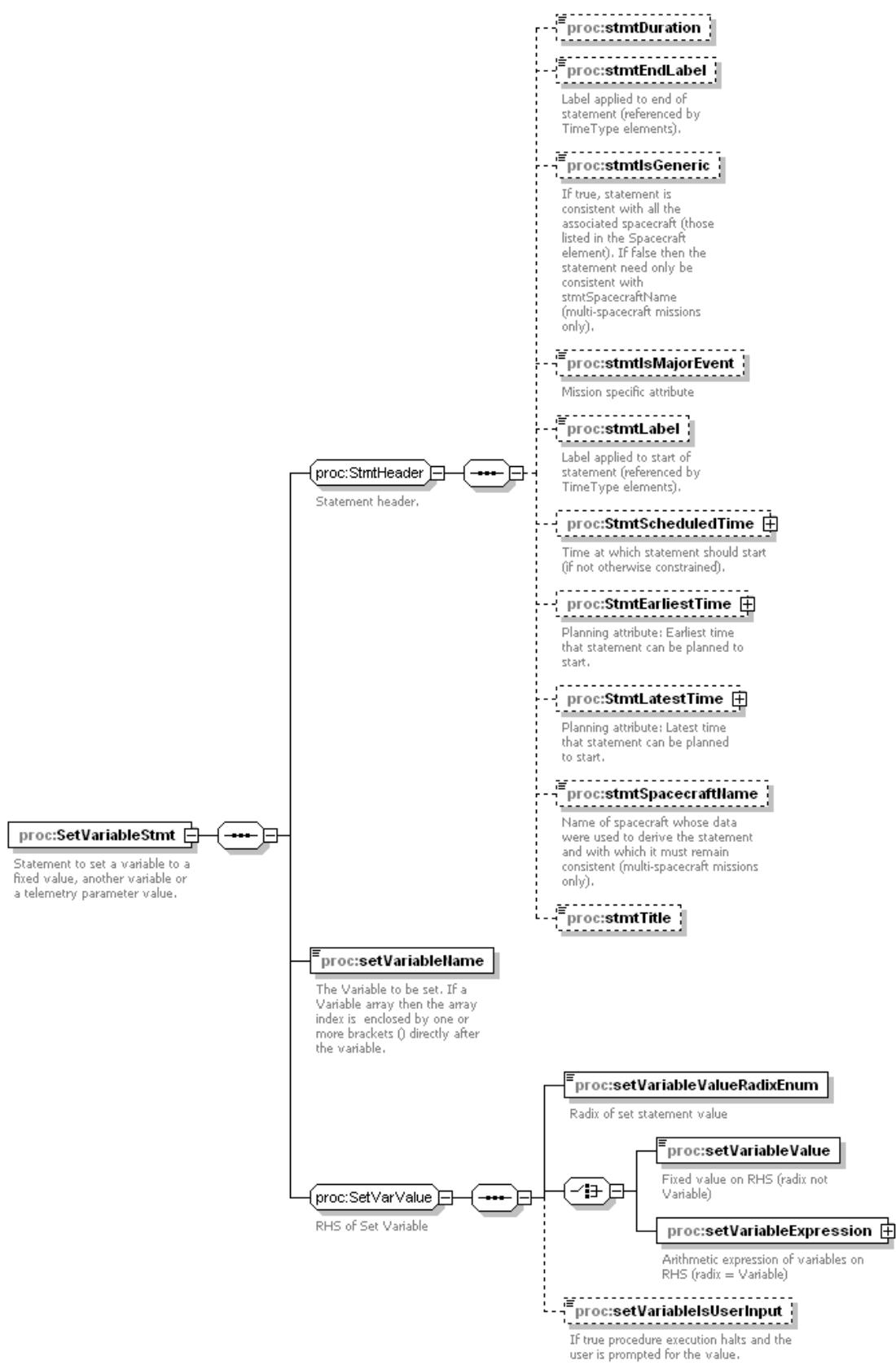
element AllStmts/DirectiveStmt/DirectiveArgument

diagram	<p>The diagram illustrates the structure of the <code>proc:DirectiveArgument</code> element. It is a class with a multiplicity of <code>0..∞</code>. It has one outgoing association named <code>proc:Parameter</code>, also with a multiplicity of <code>0..∞</code>. This association points to a class <code>proc:Parameter</code>, which has a self-loop multiplicity of <code>0..∞</code>. A note below the association indicates that it represents a "Generic parameter".</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc: paramName proc: paramDescr proc: paramEngUnit proc: paramValueRadixEnum proc: paramValue proc: paramRawValue proc: groupRepeatSize proc: ParamValueParam								
source	<pre><xs:element name="DirectiveArgument" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:group ref="proc:Parameter"/></pre>								

	</xs:complexType> </xs:element>
--	------------------------------------

element AllStmts/SetVariableStmt

diagram



namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:setVariableName proc:setVariableValueRadixEnum proc:setVariableValue proc:setVariableExpression proc:setVariablesIsUserInput
annotation	documentation Statement to set a variable to a fixed value, another variable or a telemetry parameter value.
source	<pre><xs:element name="SetVariableStmt"> <xs:annotation> <xs:documentation>Statement to set a variable to a fixed value, another variable or a telemetry parameter value.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="setVariableName" type="xs:string"> <xs:annotation> <xs:documentation>The Variable to be set. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> <xs:group ref="proc:SetVarValue"> <xs:annotation> <xs:documentation>RHS of Set Variable</xs:documentation> </xs:annotation> </xs:group> </xs:sequence> </xs:complexType> </xs:element></pre>

element AllStmts/SetVariableStmt/setVariableName

diagram	 <p>The Variable to be set. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The Variable to be set. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<pre><xs:element name="setVariableName" type="xs:string"> <xs:annotation> <xs:documentation>The Variable to be set. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/SetTelemetryStmt

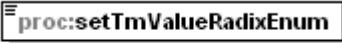
diagram	<pre> classDiagram class proc:StmtHeader { <<Statement header.>> } class proc:SetTelemetryStmt { <<Statement to set a TM parameter to an engineered value.>> } proc:StmtHeader < -- proc:SetTelemetryStmt proc:StmtHeader --> proc:stmtDuration proc:StmtHeader --> proc:stmtEndLabel proc:StmtHeader --> proc:stmtIsGeneric proc:StmtHeader --> proc:stmtMajorEvent proc:StmtHeader --> proc:stmtLabel proc:StmtHeader --> proc:stmtScheduledTime proc:StmtHeader --> proc:stmtEarliestTime proc:StmtHeader --> proc:stmtLatestTime proc:StmtHeader --> proc:stmtSpacecraftName proc:StmtHeader --> proc:stmtTitle </pre> <p>The diagram illustrates the UML class structure for the <code>SetTelemetryStmt</code> element. It inherits from the <code>StmtHeader</code> base class, which is indicated by a generalization arrow pointing from <code>StmtHeader</code> to <code>SetTelemetryStmt</code>. The <code>SetTelemetryStmt</code> class has several associations with other classes, each enclosed in a dashed box:</p> <ul style="list-style-type: none"> <code>proc:stmtDuration</code>: Label applied to end of statement (referenced by TimeType elements). <code>proc:stmtEndLabel</code>: Label applied to start of statement (referenced by TimeType elements). <code>proc:stmtIsGeneric</code>: If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If False then the statement need only be consistent with <code>stmtSpacecraftName</code> (multi-spacecraft missions only). <code>proc:stmtMajorEvent</code>: Mission specific attribute. <code>proc:stmtLabel</code>: Label applied to end of statement (referenced by TimeType elements). <code>proc:stmtScheduledTime</code>: Time at which statement should start (if not otherwise constrained). <code>proc:stmtEarliestTime</code>: Planning attribute: Earliest time that statement can be planned to start. <code>proc:stmtLatestTime</code>: Planning attribute: Latest time that statement can be planned to start. <code>proc:stmtSpacecraftName</code>: Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only). <code>proc:stmtTitle</code>: Title of the statement. <p>The <code>SetTelemetryStmt</code> class itself has a note indicating it is used to set a TM parameter to an engineered value.</p>
namespace	http://www.omg.org/space/procspec

properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:setTmParameterName proc:setTmValueRadixEnum proc:setTmValue
annotation	documentation Statement to set a TM parameter to an engineered value.
source	<pre><xs:element name="SetTelemetryStmt"> <xs:annotation> <xs:documentation>Statement to set a TM parameter to an engineered value.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="setTmParameterName" type="xs:string"> <xs:annotation> <xs:documentation>The TM Parameter to be set.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="setTmValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>Radix of set statement value </xs:documentation> </xs:annotation> </xs:element> <xs:element name="setTmValue" type="xs:string"> <xs:annotation> <xs:documentation>Fixed value on RHS (radix not Variable)</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element AllStmts/SetTelemetryStmt/setTmParameterName

diagram	 proc:setTmParameterName The TM Parameter to be set.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The TM Parameter to be set.
source	<pre><xs:element name="setTmParameterName" type="xs:string"> <xs:annotation> <xs:documentation>The TM Parameter to be set.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/SetTelemetryStmt/setTmValueRadixEnum

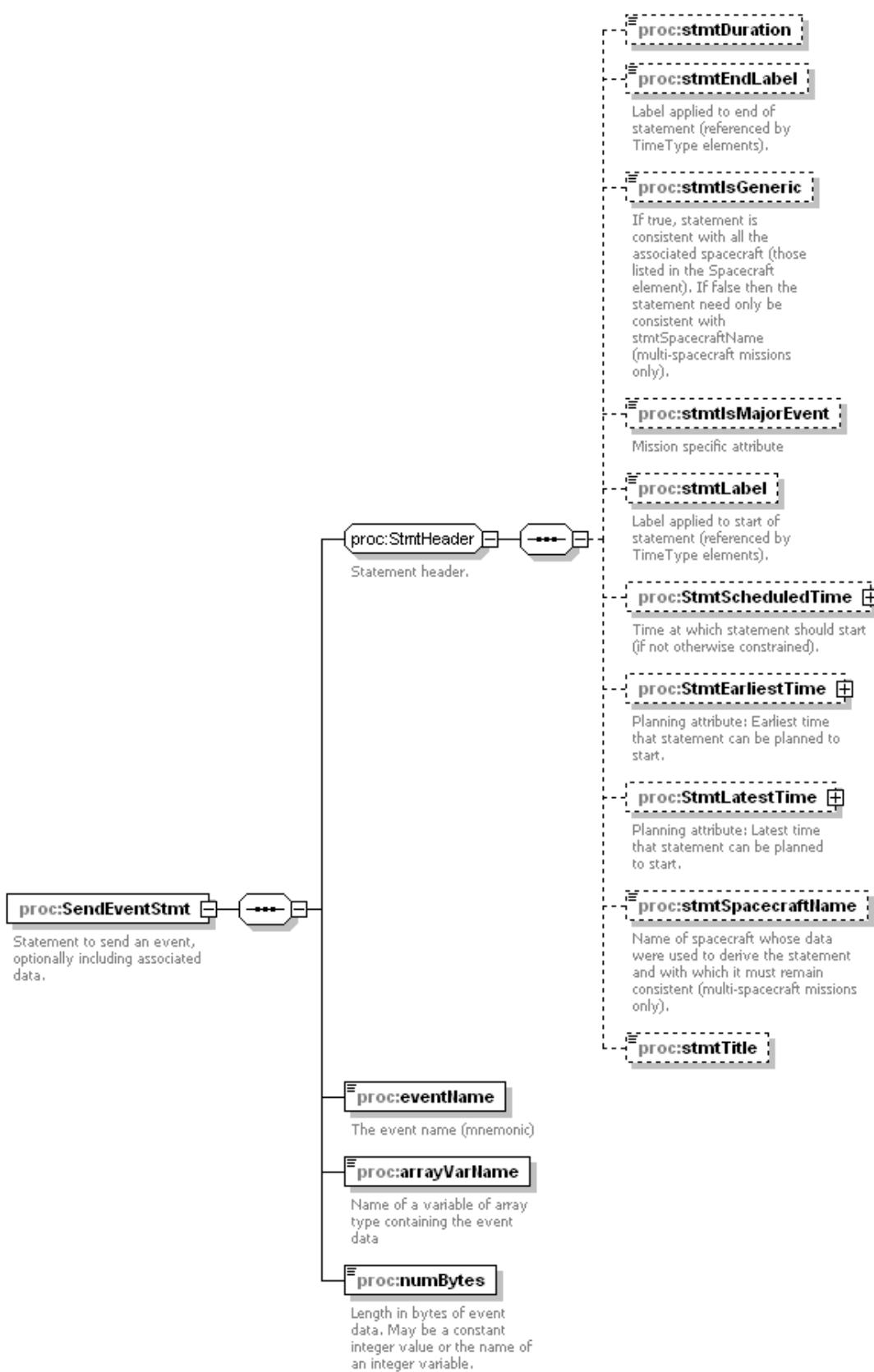
diagram	 proc:setTmValueRadixEnum Radix of set statement value
namespace	http://www.omg.org/space/procspec
type	proc:radixEnum
properties	isRef 0 content simple
annotation	documentation Radix of set statement value
source	<xs:element name="setTmValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>Radix of set statement value </xs:documentation> </xs:annotation> </xs:element>

element AllStmts/SetTelemetryStmt/setTmValue

diagram	 proc:setTmValue Fixed value on RHS (radix not Variable)
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Fixed value on RHS (radix not Variable)
source	<xs:element name="setTmValue" type="xs:string"> <xs:annotation> <xs:documentation>Fixed value on RHS (radix not Variable)</xs:documentation> </xs:annotation> </xs:element>

element AllStmts/SendEventStmt

diagram

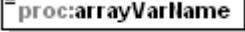


namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:eventName proc:arrayVarName proc:numBytes
annotation	documentation Statement to send an event, optionally including associated data.
source	<pre><xs:element name="SendEventStmt"> <xs:annotation> <xs:documentation>Statement to send an event, optionally including associated data.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>The event name (mnemonic)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="arrayVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of a variable of array type containing the event data</xs:documentation> </xs:annotation> </xs:element> <xs:element name="numBytes" type="xs:string"> <xs:annotation> <xs:documentation>Length in bytes of event data. May be a constant integer value or the name of an integer variable.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element AllStmts/SendEventStmt(eventName)

diagram	proc:eventName The event name (mnemonic)
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The event name (mnemonic)
source	<pre><xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>The event name (mnemonic)</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/SendEventStmt/arrayVarName

diagram	 proc:arrayVarName Name of a variable of array type containing the event data
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of a variable of array type containing the event data
source	<pre><xs:element name="arrayVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of a variable of array type containing the event data</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/SendEventStmt/numBytes

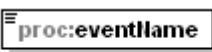
diagram	 proc:numBytes Length in bytes of event data. May be a constant integer value or the name of an integer variable.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Length in bytes of event data. May be a constant integer value or the name of an integer variable.
source	<pre><xs:element name="numBytes" type="xs:string"> <xs:annotation> <xs:documentation>Length in bytes of event data. May be a constant integer value or the name of an integer variable.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/WaitEventStmt

diagram	<p>proc:StmtHeader</p> <p>Statement header.</p> <p>proc:WaitEventStmt</p> <p>Statement to wait for an event, optionally retrieving associated data.</p> <p>proc:eventName The event name (mnemonic)</p> <p>proc:arrayVarName Name of a variable of array type to receive the event data</p> <p>proc:numBytesVarName Name of an integer variable containing the number of bytes.</p> <p>proc:timeout Optional timeout in seconds</p>	<p>proc:stmtDuration</p> <p>proc:stmtEndLabel Label applied to end of statement (referenced by TimeType elements).</p> <p>proc:stmtIsGeneric If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If false then the statement need only be consistent with stmtSpacecraftName (multi-spacecraft missions only).</p> <p>proc:stmtIsMajorEvent Mission specific attribute</p> <p>proc:stmtLabel Label applied to start of statement (referenced by TimeType elements).</p> <p>proc:StmtScheduledTime Time at which statement should start (if not otherwise constrained).</p> <p>proc:StmtEarliestTime Planning attribute: Earliest time that statement can be planned to start.</p> <p>proc:StmtLatestTime Planning attribute: Latest time that statement can be planned to start.</p> <p>proc:stmtSpacecraftName Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only).</p> <p>proc:stmtTitle</p>
---------	---	---

namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:eventName proc:arrayVarName proc:numBytesVarName proc:timeout
annotation	documentation Statement to wait for an event, optionally retrieving associated data.
source	<pre><xs:element name="WaitEventStmt"> <xs:annotation> <xs:documentation>Statement to wait for an event, optionally retrieving associated data.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>The event name (mnemonic)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="arrayVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of a variable of array type to receive the event data</xs:documentation> </xs:annotation> </xs:element> <xs:element name="numBytesVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of an integer variable containing the number of bytes.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="timeout" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Optional timeout in seconds</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element AllStmts/WaitEventStmt(eventName)

diagram	 The event name (mnemonic)
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The event name (mnemonic)

source	<pre><xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>The event name (mnemonic)</xs:documentation> </xs:annotation> </xs:element></pre>
--------	---

element AllStmts/WaitEventStmt/arrayVarName

diagram	<p>proc:arrayVarName</p> <p>Name of a variable of array type to receive the event data</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of a variable of array type to receive the event data
source	<pre><xs:element name="arrayVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of a variable of array type to receive the event data</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/WaitEventStmt/numBytesVarName

diagram	<p>proc:numBytesVarName</p> <p>Name of an integer variable containing the number of bytes.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of an integer variable containing the number of bytes.
source	<pre><xs:element name="numBytesVarName" type="xs:string"> <xs:annotation> <xs:documentation>Name of an integer variable containing the number of bytes.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/WaitEventStmt/timeout

diagram	<p>proc:timeout</p> <p>Optional timeout in seconds</p>
---------	--

namespace	http://www.omg.org/space/procspec
type	xs:long
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Optional timeout in seconds
source	<xs:element name="timeout" type="xs:long" minOccurs="0"> <xs:annotation> <xs:documentation>Optional timeout in seconds</xs:documentation> </xs:annotation> </xs:element>

element AllStmts/SeqCallStmt

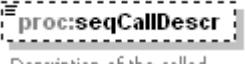
diagram	<pre> classDiagram class proc:StmtHeader class proc:SeqCallStmt { <<Statement to call another sequence.>> <<proc:SeqParam *>> <<0..>> } class proc:Stmt StmtHeader < -- SeqCallStmt Stmt < -- SeqCallStmt SeqCallStmt < -- proc:stmtDuration SeqCallStmt < -- proc:stmtEndLabel SeqCallStmt < -- proc:stmtIsGeneric SeqCallStmt < -- proc:stmtIsMajorEvent SeqCallStmt < -- proc:stmtLabel SeqCallStmt < -- proc:stmtScheduledTime SeqCallStmt < -- proc:stmtEarliestTime SeqCallStmt < -- proc:stmtLatestTime SeqCallStmt < -- proc:stmtSpacecraftName SeqCallStmt < -- proc:stmtTitle SeqParam < -- proc:seqCallName SeqParam < -- proc:seqCallDescr </pre> <p>The diagram illustrates the UML class <code>SeqCallStmt</code>. It inherits from <code>StmtHeader</code> and <code>Stmt</code>. The class <code>SeqCallStmt</code> has a multiplicity of <code>0..*</code> for its attribute <code>proc:SeqParam</code>. This attribute contains two associations: one to <code>proc:seqCallName</code> and another to <code>proc:seqCallDescr</code>.</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				

children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:seqCallName proc:seqCallDescr proc:SeqParam
annotation	documentation Statement to call another sequence.
source	<pre><xs:element name="SeqCallStmt"> <xs:annotation> <xs:documentation>Statement to call another sequence.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="seqCallName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called sequence.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="seqCallDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Description of the called sequence.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="SeqParam" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element AllStmts/SeqCallStmt/seqCallName

diagram	 <p>Name of the called sequence.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of the called sequence.
source	<pre><xs:element name="seqCallName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the called sequence.</xs:documentation> </xs:annotation> </xs:element></pre>

element **AllStmts/SeqCallStmt/seqCallDescr**

diagram	 Description of the called sequence.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Description of the called sequence.
source	<pre><xs:element name="seqCallDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Description of the called sequence.</xs:documentation> </xs:annotation> </xs:element></pre>

element AllStmts/SeqCallStmt/SeqParam

diagram	<p>The diagram illustrates the structure of the SeqParam element. It starts with a dashed box labeled proc:SeqParam with a multiplicity of 0..∞. This is followed by a solid box labeled proc:Parameter with the text "Generic parameter.". After the parameter, there is a repeating group indicated by three dots (•••) enclosed in a dashed box. Finally, there is another dashed box labeled proc:ParamValueParam with a multiplicity of 0..∞.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table border="0"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc: paramName proc: paramDescr proc: paramEngUnit proc: paramValueRadixEnum proc: paramValue proc: paramRawValue proc: groupRepeatSize proc: ParamValueParam								
source	<pre><xs:element name="SeqParam" minOccurs="0" maxOccurs="unbounded"> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType> </xs:element></pre>								

group ArithmeticComponent

diagram	<p>A component of the expression yielding an arithmetic result</p>
namespace	http://www.omg.org/space/procspec
children	proc:variableName proc:telemetry proc:fixedValue proc:TimStmt proc:CheckVariableStmt proc:PacketStmt proc:UnaryOperator proc:Bracket
used by	elements ArithmeticResult/Bracket ArithmeticResult/Bracket
annotation	documentation A component of the expression yielding an arithmetic result
source	<pre> <xs:group name="ArithmeticComponent"> <xs:annotation> <xs:documentation>A component of the expression yielding an arithmetic result</xs:documentation> </xs:annotation> <xs:sequence> <xs:choice> <xs:element name="variableName" type="xs:string"> <xs:annotation> <xs:documentation>Varable name. If a Variable array then the array index is enclosed by</pre>

	<p>one or more brackets () directly after the variable</p> <pre> </xs:documentation> </xs:annotation> </xs:element> <xs:element name="telemetry" type="xs:string"> <xs:annotation> <xs:documentation>Name of a telemetry parameter of which the current value shall be taken.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="fixedValue"> <xs:annotation> <xs:documentation>A literal value in the expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:string"> <xs:attribute name="Radix" type="proc:valueRadixEnum" use="optional" default="Decimal"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element> <xs:group ref="proc:BooleanStmts"/> <xs:group ref="proc:ArithmeticResult"/> </xs:choice> </xs:sequence> </xs:group></pre>
--	---

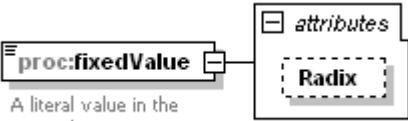
element ArithmeticComponent/variableName

diagram	 proc:variableName <p>Variable name. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	<p>documentation</p> <p>Variable name. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable</p>
source	<pre> <xs:element name="variableName" type="xs:string"> <xs:annotation> <xs:documentation>Variable name. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable</xs:documentation> </xs:annotation> </xs:element></pre>

element ArithmeticComponent/telemetry

diagram	 Name of a telemetry parameter of which the current value shall be taken.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of a telemetry parameter of which the current value shall be taken.
source	<pre><xs:element name="telemetry" type="xs:string"> <xs:annotation> <xs:documentation>Name of a telemetry parameter of which the current value shall be taken.</xs:documentation> </xs:annotation> </xs:element></pre>

element ArithmeticComponent/fixedValue

diagram	 A literal value in the expression												
namespace	http://www.omg.org/space/procspec												
type	extension of xs:string												
properties	isRef 0 content complex												
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>annotation</th> </tr> </thead> <tbody> <tr> <td>Radix</td> <td>proc:valueRadixEnum</td> <td>optional</td> <td>Decimal</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	annotation	Radix	proc:valueRadixEnum	optional	Decimal		
Name	Type	Use	Default	Fixed	annotation								
Radix	proc:valueRadixEnum	optional	Decimal										
annotation	documentation A literal value in the expression												
source	<pre><xs:element name="fixedValue"> <xs:annotation> <xs:documentation>A literal value in the expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:simpleContent> <xs:extension base="xs:string"> <xs:attribute name="Radix" type="proc:valueRadixEnum" use="optional" default="Decimal"/> </xs:extension> </xs:simpleContent> </xs:complexType> </xs:element></pre>												

attribute ArithmeticComponent/fixedValue/@Radix

type	proc:valueRadixEnum
properties	isRef 0 default Decimal

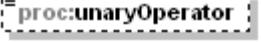
	use optional
facets	enumeration Decimal enumeration Hexadecimal enumeration Octal enumeration Binary
source	<xs:attribute name="Radix" type="proc:valueRadixEnum" use="optional" default="Decimal"/>

group ArithmeticResult

diagram	<p>The arithmetic result of an expression.</p>
namespace	http://www.omg.org/space/procspec
children	proc:unaryOperator proc:Bracket
used by	elements StepType/DecisionStep/ArithmeticResult StepType/SWITCHstep/Case/CaseArithmeticResult SetVarValue/setVariableExpression group ArithmeticComponent
annotation	documentation The arithmetic result of an expression.
source	<xs:group name="ArithmeticResult"> <xs:annotation> <xs:documentation>The arithmetic result of an expression.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="unaryOperator" type="proc:unaryOperator" minOccurs="0"> <xs:annotation> <xs:documentation>Applies a unary operator minus (-) or NOT to the bracketed expression</xs:documentation> </xs:annotation> </xs:element> <xs:element name="Bracket"> <xs:annotation> <xs:documentation>Brackets the following expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:ArithmeticComponent"/> <xs:sequence minOccurs="0"> <xs:element name="binaryOperator" type="proc:binaryOperator"> <xs:annotation> <xs:documentation>Binary operator</xs:documentation> </xs:annotation> </xs:element> <xs:group ref="proc:ArithmeticComponent"/> </xs:sequence> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence>

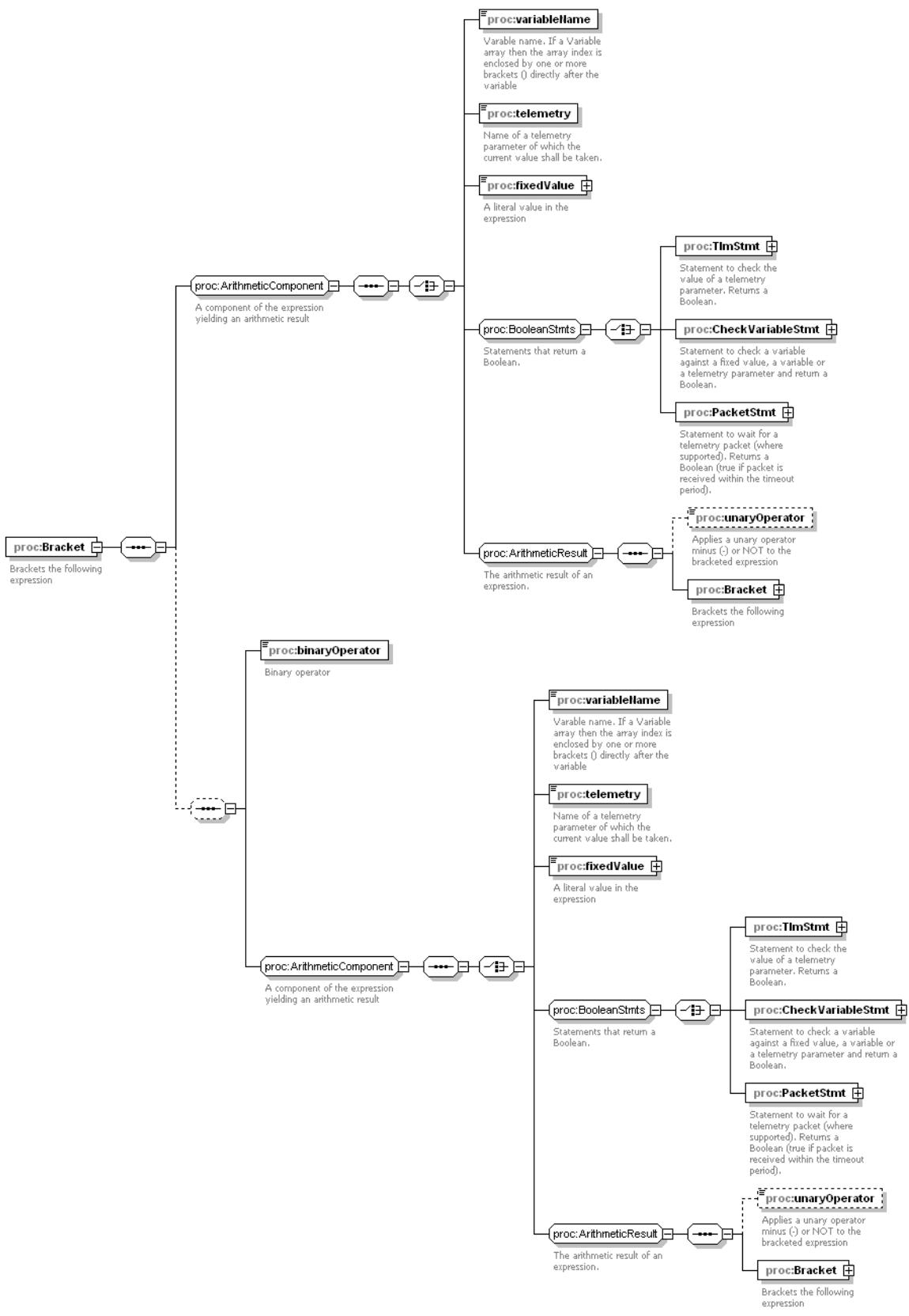
	</xs:group>
--	-------------

element **ArithmeticResult/unaryOperator**

diagram	 <p>Applies a unary operator minus (-) or NOT to the bracketed expression</p>								
namespace	http://www.omg.org/space/procspec								
type	proc:unaryOperator								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
facets	<table> <tr> <td>enumeration</td><td>-</td></tr> <tr> <td>enumeration</td><td>NOT</td></tr> </table>	enumeration	-	enumeration	NOT				
enumeration	-								
enumeration	NOT								
annotation	<p>documentation</p> <p>Applies a unary operator minus (-) or NOT to the bracketed expression</p>								
source	<pre><xs:element name="unaryOperator" type="proc:unaryOperator" minOccurs="0"> <xs:annotation> <xs:documentation>Applies a unary operator minus (-) or NOT to the bracketed expression</xs:documentation> </xs:annotation> </xs:element></pre>								

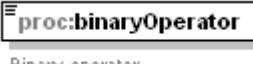
element ArithmeticResult/Bracket

diagram



namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:variableName proc:telemetry proc:fixedValue proc:TImStmt proc:CheckVariableStmt proc:PacketStmt proc:unaryOperator proc:Bracket proc:binaryOperator
annotation	documentation Brackets the following expression
source	<pre><xs:element name="Bracket"> <xs:annotation> <xs:documentation>Brackets the following expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:ArithmeticComponent"/> <xs:sequence minOccurs="0"> <xs:element name="binaryOperator" type="proc:binaryOperator"> <xs:annotation> <xs:documentation>Binary operator</xs:documentation> </xs:annotation> </xs:element> <xs:group ref="proc:ArithmeticComponent"/> </xs:sequence> </xs:sequence> </xs:complexType> </xs:element></pre>

element ArithmeticResult/Bracket/binaryOperator

diagram	 proc:binaryOperator Binary operator
namespace	http://www.omg.org/space/procspec
type	proc:binaryOperator
properties	isRef 0 content simple
facets	enumeration + enumeration - enumeration * enumeration / enumeration ^ enumeration AND enumeration OR enumeration XOR enumeration == enumeration NE enumeration LT enumeration LE enumeration GT enumeration GE enumeration LEFTSHIFT enumeration RIGHTSHIFT enumeration BAND enumeration BOR enumeration BXOR enumeration CONCAT enumeration MOD
annotation	documentation Binary operator

source	<pre><xs:element name="binaryOperator" type="proc:binaryOperator"> <xs:annotation> <xs:documentation>Binary operator</xs:documentation> </xs:annotation> </xs:element></pre>
group BooleanComponent	<p>diagram</p>
namespace	http://www.omg.org/space/procspec
children	proc:TimStmt proc:CheckVariableStmt proc:PacketStmt proc:NOT proc:Bracket
used by	elements BooleanResult/Bracket BooleanResult/Bracket
annotation	documentation A component of the expression yeilding a Boolean result
source	<pre><xs:group name="BooleanComponent"> <xs:annotation> <xs:documentation>A component of the expression yeilding a Boolean result</xs:documentation> </xs:annotation> <xs:sequence> <xs:choice> <xs:group ref="proc:BooleanStmts"/> <xs:group ref="proc:BooleanResult"/> </xs:choice> </xs:sequence> </xs:group></pre>

group BooleanResult

diagram	<p>The Boolean result of an expression.</p> <p>proc:NOT NOTs the bracketed expression</p> <p>proc:Bracket Brackets the following expression</p>
namespace	http://www.omg.org/space/procspec
children	proc:NOT proc:Bracket
used by	elements StepType/DecisionStep/BooleanResult StepType/SWITCHstep/Case/CaseBooleanResult group BooleanComponent
annotation	documentation The Boolean result of an expression.
source	<pre><xs:group name="BooleanResult"> <xs:annotation> <xs:documentation>The Boolean result of an expression.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="NOT" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>NOTs the bracketed expression</xs:documentation> </xs:annotation> </xs:element> <xs:element name="Bracket"> <xs:annotation> <xs:documentation>Brackets the following expression</xs:documentation> </xs:annotation> </xs:element> <xs:complexType> <xs:sequence> <xs:group ref="proc:BooleanComponent"/> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element name="booleanOperator" type="proc:booleanOperator"> <xs:annotation> <xs:documentation>AND, OR or XOR</xs:documentation> </xs:annotation> </xs:element> <xs:group ref="proc:BooleanComponent"/> </xs:sequence> </xs:complexType> </xs:sequence> </xs:group></pre>

element BooleanResult/NOT

diagram	<p>NOTs the bracketed expression</p>
namespace	http://www.omg.org/space/procspec

type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple default false
annotation	documentation NOTs the bracketed expression
source	<pre><xs:element name="NOT" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>NOTs the bracketed expression</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanResult/Bracket

diagram	
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:TimStmt proc:CheckVariableStmt proc:PacketStmt proc:NOT proc:Bracket proc:booleanOperator

annotation	documentation Brackets the following expression
source	<pre><xs:element name="Bracket"> <xs:annotation> <xs:documentation>Brackets the following expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:BooleanComponent"/> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:element name="booleanOperator" type="proc:booleanOperator"> <xs:annotation> <xs:documentation>AND, OR or XOR</xs:documentation> </xs:annotation> </xs:element> <xs:group ref="proc:BooleanComponent"/> </xs:sequence> </xs:sequence> </xs:complexType> </xs:element></pre>

element BooleanResult/Bracket/booleanOperator

diagram	 AND, OR or XOR
namespace	http://www.omg.org/space/procspec
type	proc:booleanOperator
properties	isRef 0 content simple
facets	enumeration AND enumeration OR enumeration XOR
annotation	documentation AND, OR or XOR
source	<pre><xs:element name="booleanOperator" type="proc:booleanOperator"> <xs:annotation> <xs:documentation>AND, OR or XOR</xs:documentation> </xs:annotation> </xs:element></pre>

group BooleanStmts

diagram	<pre> sequenceDiagram participant BS as BooleanStmts participant T as TlmStmt participant CV as CheckVariableStmt participant P as PacketStmt BS->>T: activate T T-->>CV: activate CV CV-->>P: deactivate CV deactivate T </pre> <p>Statements that return a Boolean.</p> <p>proc:TlmStmt</p> <p>Statement to check the value of a telemetry parameter. Returns a Boolean.</p> <p>proc:CheckVariableStmt</p> <p>Statement to check a variable against a fixed value, a variable or a telemetry parameter and return a Boolean.</p> <p>proc:PacketStmt</p> <p>Statement to wait for a telemetry packet (where supported). Returns a Boolean (true if packet is received within the timeout period).</p>
namespace	http://www.omg.org/space/procspec
children	proc:TlmStmt proc:CheckVariableStmt proc:PacketStmt
used by	groups AllStmts ArithmeticComponent BooleanComponent
annotation	documentation Statements that return a Boolean.
source	<pre> <xs:group name="BooleanStmts"> <xs:annotation> <xs:documentation>Statements that return a Boolean.</xs:documentation> </xs:annotation> <xs:choice> <xs:element name="TlmStmt"> <xs:annotation> <xs:documentation>Statement to check the value of a telemetry parameter. Returns a Boolean.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="tlmName" type="xs:string"> <xs:annotation> <xs:documentation>The name of the telemetry parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmPid" type="xs:unsignedInt" minOccurs="0"> <xs:annotation> <xs:documentation>The PID of the telemetry parameter or its enclosing global if it is a detailed parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmType" type="proc:variableTypeEnum" minOccurs="0"> <xs:annotation> <xs:documentation>The value type of the telemetry parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmOffset" type="xs:unsignedShort" minOccurs="0"> </pre>

```

<xs:annotation>
  <xs:documentation>The offset of the TLM parameter in its enclosing global, counted from
the left (MSB).</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="tlmLength" type="xs:unsignedShort" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The number of bits of this TLM parameter.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmDescr" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The description of the telemetry parameter</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmDetailedDescr" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The detailed description of the telemetry
parameter</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmEngUnit" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The telemetry parameter engineering unit</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmGlobalLength" type="xs:unsignedShort" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The number of bits of the enclosing global TLM
parameter.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmCmdRef" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The associated command reference if this is its Pre-Transmission-
Validation (PTV) or Command-Execution-Verification (CEV) telemetry check, in the form stmtTitle
cmdName.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmPacketRef" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The associated packet reference if this telemetry must be read from
the same packet, in the form stmtTitle packetName.</xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="tlmDisplayRef" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Display reference of one or more TM Parameters belonging to the
original FOP.</xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="tlmSelectedCalcurve" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The name of the selected calibration curve (if more than 1 is defined

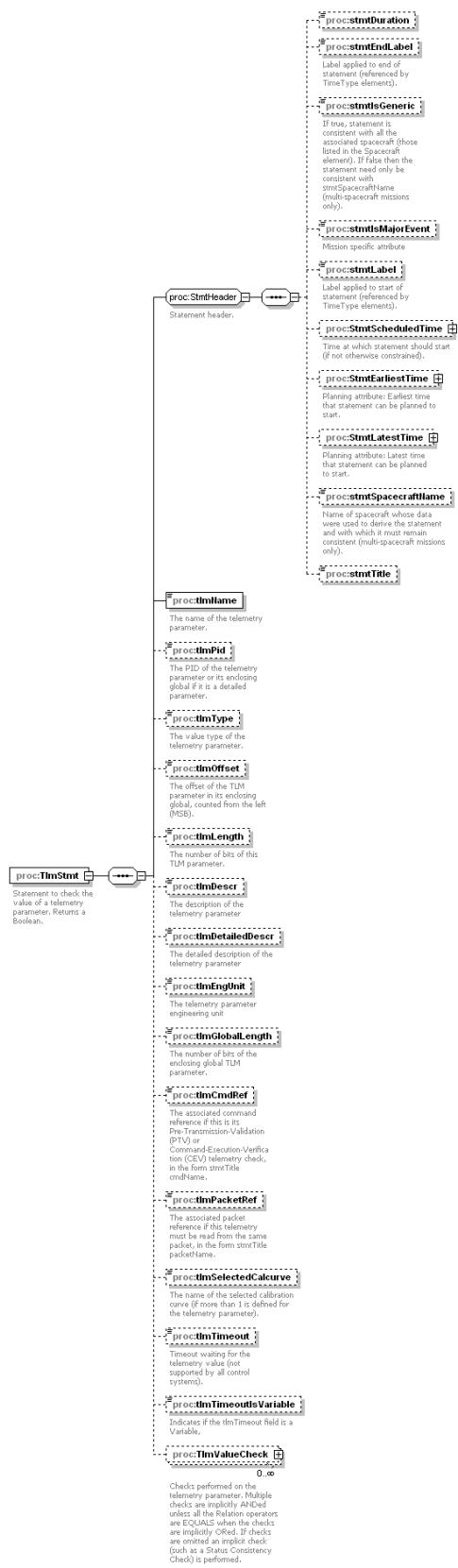
```

	<p>for the telemetry parameter).</xs:documentation></p> <p></xs:annotation></p> <p></xs:element></p> <p><xs:element name="tlmTimeout" type="xs:string" minOccurs="0"></p> <p><xs:annotation></p> <p><xs:documentation>Timeout waiting for the telemetry value (not supported by all control systems).</xs:documentation></p> <p></xs:annotation></p> <p></xs:element></p> <p><xs:element name="tlmTimeoutIsVariable" type="xs:boolean" default="false" minOccurs="0"></p> <p><xs:annotation></p> <p><xs:documentation>Indicates if the tlmTimeout field is a Variable,</xs:documentation></p> <p></xs:annotation></p> <p></xs:element></p> <p><xs:element name="TlmValueCheck" minOccurs="0" maxOccurs="unbounded"></p> <p><xs:annotation></p> <p><xs:documentation>Checks performed on the telemetry parameter. Multiple checks are implicitly ANDed unless all the Relation operators are EQUALS when the checks are implicitly ORed. If checks are omitted an implicit check (such as a Status Consistency Check) is performed.</xs:documentation></p> <p></xs:annotation></p> <p><xs:complexType></p> <p><xs:sequence></p> <p><xs:element name="tlmValueRelationEnum" type="proc:relationEnum"/></p> <p><xs:element name="tlmValueRadixEnum" type="proc:radixEnum"/></p> <p><xs:element name="tlmValue" type="xs:string"></p> <p><xs:annotation></p> <p><xs:documentation>The value of the TM parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation></p> <p></xs:annotation></p> <p></xs:element></p> <p></xs:sequence></p> <p></xs:complexType></p> <p></xs:element></p> <p><xs:element name="CheckVariableStmt"></p> <p><xs:annotation></p> <p><xs:documentation>Statement to check a variable against a fixed value, a variable or a telemetry parameter and return a Boolean.</xs:documentation></p> <p></xs:annotation></p> <p><xs:complexType></p> <p><xs:sequence></p> <p><xs:group ref="proc:StmtHeader"/></p> <p><xs:element name="checkVariableName" type="xs:string"></p> <p><xs:annotation></p> <p><xs:documentation>Name of Variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation></p> <p></xs:annotation></p> <p></xs:element></p> <p><xs:group ref="proc:CheckVarValue"/></p> <p></xs:sequence></p> <p></xs:complexType></p>
--	--

```
</xs:element>
<xs:element name="PacketStmt">
<xs:annotation>
<xs:documentation>Statement to wait for a telemetry packet (where supported). Returns a Boolean (true if packet is received within the timeout period).</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:group ref="proc:StmtHeader"/>
<xs:element name="packetName" type="xs:string">
<xs:annotation>
<xs:documentation>Name of the telemetry packet.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="packetDescr" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Description of the telemetry packet.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="packetTimeout" type="xs:duration" minOccurs="0">
<xs:annotation>
<xs:documentation>Timeout waiting for the packet.</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:group>
```

element BooleanStmts/TlmStmt

diagram



namespace	http://www.omg.org/space/procspec				
properties	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="padding: 2px;">isRef</td><td style="padding: 2px;">0</td></tr> <tr> <td style="padding: 2px;">content</td><td style="padding: 2px;">complex</td></tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	<p style="margin: 0;">proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel</p> <p style="margin: 0;">proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle</p> <p style="margin: 0;">proc:tlmName proc:tlmPid proc:tlmType proc:tlmOffset proc:tlmLength proc:tlmDescr proc:tlmDetailedDescr</p> <p style="margin: 0;">proc:tlmEngUnit proc:tlmGlobalLength proc:tlmCmdRef proc:tlmPacketRef proc:tlmDisplayRef</p> <p style="margin: 0;">proc:tlmSelectedCalcurve proc:tlmTimeoutsVariable proc:TlmValueCheck</p>				
annotation	documentation Statement to check the value of a telemetry parameter. Returns a Boolean.				
source	<pre style="font-family: monospace; font-size: 0.8em; margin: 0; padding: 0;"> <xs:element name="TlmStmt"> <xs:annotation> <xs:documentation>Statement to check the value of a telemetry parameter. Returns a Boolean.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="tlmName" type="xs:string"> <xs:annotation> <xs:documentation>The name of the telemetry parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmPid" type="xs:unsignedInt" minOccurs="0"> <xs:annotation> <xs:documentation>The PID of the telemetry parameter or its enclosing global if it is a detailed parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmType" type="proc:variableTypeEnum" minOccurs="0"> <xs:annotation> <xs:documentation>The value type of the telemetry parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmOffset" type="xs:unsignedShort" minOccurs="0"> <xs:annotation> <xs:documentation>The offset of the TLM parameter in its enclosing global, counted from the left (MSB).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmLength" type="xs:unsignedShort" minOccurs="0"> <xs:annotation> <xs:documentation>The number of bits of this TLM parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The description of the telemetry parameter</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmDetailedDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The detailed description of the telemetry parameter</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tlmEngUnit" type="xs:string" minOccurs="0"> <xs:annotation> </pre>				

```

<xs:annotation>
  <xs:documentation>The telemetry parameter engineering unit</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="tlmGlobalLength" type="xs:unsignedShort" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The number of bits of the enclosing global TLM
parameter.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmCmdRef" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The associated command reference if this is its Pre-Transmission-
Validation (PTV) or Command-Execution-Verification (CEV) telemetry check, in the form stmtTitle
cmdName.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmPacketRef" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The associated packet reference if this telemetry must be read from the
same packet, in the form stmtTitle packetName.</xs:documentation>
  </xs:annotation>
</xs:element>

<xs:element name="tlmDisplayRef" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Display reference of one or more TM Parameters belonging to the
original FOP.</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmSelectedCalcurve" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>The name of the selected calibration curve (if more than 1 is defined for
the telemetry parameter).</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmTimeout" type="xs:string" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Timeout waiting for the telemetry value (not supported by all control
systems).</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="tlmTimeoutIsVariable" type="xs:boolean" default="false" minOccurs="0">
  <xs:annotation>
    <xs:documentation>Indicates if the tlmTimeout field is a Variable,</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="TlmValueCheck" minOccurs="0" maxOccurs="unbounded">
  <xs:annotation>
    <xs:documentation>Checks performed on the telemetry parameter. Multiple checks are
implicitly ANDed unless all the Relation operators are EQUALS when the checks are implicitly
ORed. If checks are omitted an implicit check (such as a Status Consistency Check) is
performed.</xs:documentation>
  </xs:annotation>
<xs:complexType>
  <xs:sequence>

```

	<pre> <xs:element name="tlmValueRelationEnum" type="proc:relationEnum"/> <xs:element name="tlmValueRadixEnum" type="proc:radixEnum"/> <xs:element name="tlmValue" type="xs:string"> <xs:annotation> <xs:documentation>The value of the TM parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> <xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	---

element BooleanStmts/TImStmt/tlmName

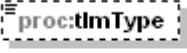
diagram	<p>The name of the telemetry parameter.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The name of the telemetry parameter.
source	<pre> <xs:element name="tlmName" type="xs:string"> <xs:annotation> <xs:documentation>The name of the telemetry parameter.</xs:documentation> </xs:annotation> </xs:element> </pre>

element BooleanStmts/TImStmt/tlmPid

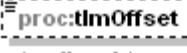
diagram	<p>The PID of the telemetry parameter or its enclosing global if it is a detailed parameter.</p>
namespace	http://www.omg.org/space/procspec
type	xs:unsignedInt
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The PID of the telemetry parameter or its enclosing global if it is a detailed parameter.
source	<pre> <xs:element name="tlmPid" type="xs:unsignedInt" minOccurs="0"> <xs:annotation> <xs:documentation>The PID of the telemetry parameter or its enclosing global if it is a detailed </pre>

	parameter.</xs:documentation> </xs:annotation> </xs:element>
--	--

element BooleanStmts/TlmStmt/tlmType

diagram	 The value type of the telemetry parameter.																										
namespace	http://www.omg.org/space/procspec																										
type	proc:variableTypeEnum																										
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple																		
isRef	0																										
minOcc	0																										
maxOcc	1																										
content	simple																										
facets	<table> <tr><td>enumeration</td><td>DERIVED</td></tr> <tr><td>enumeration</td><td>BOOLEAN</td></tr> <tr><td>enumeration</td><td>BYTE</td></tr> <tr><td>enumeration</td><td>INTEGER</td></tr> <tr><td>enumeration</td><td>LONG_INTEGER</td></tr> <tr><td>enumeration</td><td>UNSIGNED_BYTE</td></tr> <tr><td>enumeration</td><td>UNSIGNED_INTEGER</td></tr> <tr><td>enumeration</td><td>UNSIGNED_LONG_INTEGER</td></tr> <tr><td>enumeration</td><td>REAL</td></tr> <tr><td>enumeration</td><td>DOUBLE</td></tr> <tr><td>enumeration</td><td>RELATIVETIME</td></tr> <tr><td>enumeration</td><td>ABSOLUTETIME</td></tr> <tr><td>enumeration</td><td>STRING</td></tr> </table>	enumeration	DERIVED	enumeration	BOOLEAN	enumeration	BYTE	enumeration	INTEGER	enumeration	LONG_INTEGER	enumeration	UNSIGNED_BYTE	enumeration	UNSIGNED_INTEGER	enumeration	UNSIGNED_LONG_INTEGER	enumeration	REAL	enumeration	DOUBLE	enumeration	RELATIVETIME	enumeration	ABSOLUTETIME	enumeration	STRING
enumeration	DERIVED																										
enumeration	BOOLEAN																										
enumeration	BYTE																										
enumeration	INTEGER																										
enumeration	LONG_INTEGER																										
enumeration	UNSIGNED_BYTE																										
enumeration	UNSIGNED_INTEGER																										
enumeration	UNSIGNED_LONG_INTEGER																										
enumeration	REAL																										
enumeration	DOUBLE																										
enumeration	RELATIVETIME																										
enumeration	ABSOLUTETIME																										
enumeration	STRING																										
annotation	<p>documentation The value type of the telemetry parameter.</p>																										
source	<pre><xs:element name="tlmType" type="proc:variableTypeEnum" minOccurs="0"> <xs:annotation> <xs:documentation>The value type of the telemetry parameter.</xs:documentation> </xs:annotation> </xs:element></pre>																										

element BooleanStmts/TlmStmt/tlmOffset

diagram	 The offset of the TLM parameter in its enclosing global, counted from the left (MSB).								
namespace	http://www.omg.org/space/procspec								
type	xs:unsignedShort								
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation The offset of the TLM parameter in its enclosing global, counted from the left (MSB).</p>								
source	<pre><xs:element name="tlmOffset" type="xs:unsignedShort" minOccurs="0"> <xs:annotation> <xs:documentation>The offset of the TLM parameter in its enclosing global, counted from the left</pre>								

	(MSB).</xs:documentation> </xs:annotation> </xs:element>
--	--

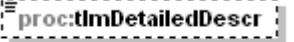
element BooleanStmts/TlmStmt/tlmLength

diagram	 proc:tlmLength The number of bits of this TLM parameter.
namespace	http://www.omg.org/space/procspec
type	xs:unsignedShort
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The number of bits of this TLM parameter.
source	<xs:element name="tlmLength" type="xs:unsignedShort" minOccurs="0"> <xs:annotation> <xs:documentation>The number of bits of this TLM parameter.</xs:documentation> </xs:annotation> </xs:element>

element BooleanStmts/TlmStmt/tlmDescr

diagram	 proc:tlmDescr The description of the telemetry parameter
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The description of the telemetry parameter
source	<xs:element name="tlmDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The description of the telemetry parameter</xs:documentation> </xs:annotation> </xs:element>

element BooleanStmts/TlmStmt/tlmDetailedDescr

diagram	 proc:tlmDetailedDescr The detailed description of the telemetry parameter
namespace	http://www.omg.org/space/procspec

type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The detailed description of the telemetry parameter
source	<pre><xs:element name="tlmDetailedDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The detailed description of the telemetry parameter</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanStmts/TlmStmt/tlmEngUnit

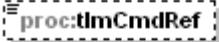
diagram	proc:tlmEngUnit The telemetry parameter engineering unit
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The telemetry parameter engineering unit
source	<pre><xs:element name="tlmEngUnit" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The telemetry parameter engineering unit</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanStmts/TlmStmt/tlmGlobalLength

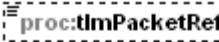
diagram	proc:tlmGlobalLength The number of bits of the enclosing global TLM parameter.
namespace	http://www.omg.org/space/procspec
type	xs:unsignedShort
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The number of bits of the enclosing global TLM parameter.
source	<pre><xs:element name="tlmGlobalLength" type="xs:unsignedShort" minOccurs="0"> <xs:annotation> <xs:documentation>The number of bits of the enclosing global TLM parameter.</xs:documentation> </xs:annotation></pre>

	</xs:element>
--	---------------

element BooleanStmts/TlmStmt/tlmCmdRef

diagram	 <p>The associated command reference if this is its Pre-Transmission-Validation (PTV) or Command-Execution-Verification (CEV) telemetry check, in the form stmtTitle cmdName.</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>The associated command reference if this is its Pre-Transmission-Validation (PTV) or Command-Execution-Verification (CEV) telemetry check, in the form stmtTitle cmdName.</p>								
source	<pre><xs:element name="tlmCmdRef" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The associated command reference if this is its Pre-Transmission-Validation (PTV) or Command-Execution-Verification (CEV) telemetry check, in the form stmtTitle cmdName.</xs:documentation> </xs:annotation> </xs:element></pre>								

element BooleanStmts/TlmStmt/tlmPacketRef

diagram	 <p>The associated packet reference if this telemetry must be read from the same packet, in the form stmtTitle packetName.</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>The associated packet reference if this telemetry must be read from the same packet, in the form stmtTitle packetName.</p>								
source	<pre><xs:element name="tlmPacketRef" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The associated packet reference if this telemetry must be read from the same packet, in the form stmtTitle packetName.</xs:documentation> </xs:annotation> </xs:element></pre>								

element BooleanStmts/TlmStmt/tlmDisplayRef

diagram	tlmDisplayRef
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The associated packet reference if this telemetry must be read from the same packet, in the form stmtTitle packetName.
source	<pre><xs:element name="tlmDisplayRef" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Display reference of one or more TM Parameters belonging to the original FOP.</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanStmts/TlmStmt/tlmSelectedCalcurve

diagram	<p>The name of the selected calibration curve (if more than 1 is defined for the telemetry parameter).</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The name of the selected calibration curve (if more than 1 is defined for the telemetry parameter).
source	<pre><xs:element name="tlmSelectedCalcurve" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>The name of the selected calibration curve (if more than 1 is defined for the telemetry parameter).</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanStmts/TlmStmt/tlmTimeout

diagram	<p>Timeout waiting for the telemetry value (not supported by all control systems).</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple

annotation	documentation Timeout waiting for the telemetry value (not supported by all control systems).
source	<pre><xs:element name="tlmTimeout" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout waiting for the telemetry value (not supported by all control systems).</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanStmts/TlmStmt/tlmTimeoutIsVariable

diagram	<p>proc:tlmTimeoutIsVariable</p> <p>Indicates if the tlmTimeout field is a Variable,</p>										
namespace	http://www.omg.org/space/procspec										
type	xs:boolean										
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>simple</td></tr> <tr> <td>default</td><td>false</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	false
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	false										
annotation	documentation Indicates if the tlmTimeout field is a Variable,										
source	<pre><xs:element name="tlmTimeoutIsVariable" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>Indicates if the tlmTimeout field is a Variable,</xs:documentation> </xs:annotation> </xs:element></pre>										

element BooleanStmts/TlmStmt/TlmValueCheck

diagram	<p>proc:TlmValueCheck</p> <p>0..oo</p> <p>Checks performed on the telemetry parameter. Multiple checks are implicitly ANDed unless all the Relation operators are EQUALS when the checks are implicitly ORed. If checks are omitted an implicit check (such as a Status Consistency Check) is performed.</p> <p>proc:tlmValueRelationEnum</p> <p>proc:tlmValueRadixEnum</p> <p>proc:tlmValue</p> <p>The value of the TM parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc:tlmValueRelationEnum proc:tlmValueRadixEnum proc:tlmValue								
annotation	documentation Checks performed on the telemetry parameter. Multiple checks are implicitly ANDed unless all the Relation operators are EQUALS when the checks are implicitly ORed. If checks are omitted an implicit check (such as a Status Consistency Check) is performed.								

source	<pre> <xs:element name="TlmValueCheck" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Checks performed on the telemetry parameter. Multiple checks are implicitly ANDed unless all the Relation operators are EQUALS when the checks are implicitly ORed. If checks are omitted an implicit check (such as a Status Consistency Check) is performed.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="tlmValueRelationEnum" type="proc:relationEnum"/> <xs:element name="tlmValueRadixEnum" type="proc:radixEnum"/> <xs:element name="tlmValue" type="xs:string"> <xs:annotation> <xs:documentation>The value of the TM parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--------	---

element BooleanStmts/TlmStmt/TlmValueCheck/tlmValueRelationEnum

diagram													
namespace	http://www.omg.org/space/procspec												
type	proc:relationEnum												
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple								
isRef	0												
content	simple												
facets	<table> <tr> <td>enumeration</td> <td>EQUALS</td> </tr> <tr> <td>enumeration</td> <td>NOTEQUALS</td> </tr> <tr> <td>enumeration</td> <td>LESSTHAN</td> </tr> <tr> <td>enumeration</td> <td>LESSTHANOEQUALS</td> </tr> <tr> <td>enumeration</td> <td>GREATERTHAN</td> </tr> <tr> <td>enumeration</td> <td>GREATERTHANOEQUALS</td> </tr> </table>	enumeration	EQUALS	enumeration	NOTEQUALS	enumeration	LESSTHAN	enumeration	LESSTHANOEQUALS	enumeration	GREATERTHAN	enumeration	GREATERTHANOEQUALS
enumeration	EQUALS												
enumeration	NOTEQUALS												
enumeration	LESSTHAN												
enumeration	LESSTHANOEQUALS												
enumeration	GREATERTHAN												
enumeration	GREATERTHANOEQUALS												
source	<xs:element name="tlmValueRelationEnum" type="proc:relationEnum"/>												

element BooleanStmts/TlmStmt/TlmValueCheck/tlmValueRadixEnum

diagram					
namespace	http://www.omg.org/space/procspec				
type	proc:radixEnum				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
source	<xs:element name="tlmValueRadixEnum" type="proc:radixEnum"/>				

element BooleanStmts/TImStmt/TImValueCheck/tlmValue

diagram	
	The value of the TM parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The value of the TM parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<pre><xs:element name="tImValue" type="xs:string"> <xs:annotation> <xs:documentation>The value of the TM parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanStmts/CheckVariableStmt

diagram	<pre> classDiagram proc:CheckVariableStmt < -- proc:StatementHeader proc:CheckVariableStmt < -- proc:checkVariableName proc:CheckVariableStmt "1..>" --- proc:CheckVarValue </pre> <p>The diagram illustrates the UML class structure for <code>proc:CheckVariableStmt</code>. It inherits from <code>proc:StatementHeader</code> and also has a dependency on <code>proc:checkVariableName</code>. Additionally, it aggregates multiple instances of <code>proc:CheckVarValue</code>, indicated by a multiplicity of <code>1..></code>.</p>
namespace	http://www.omg.org/space/procspec

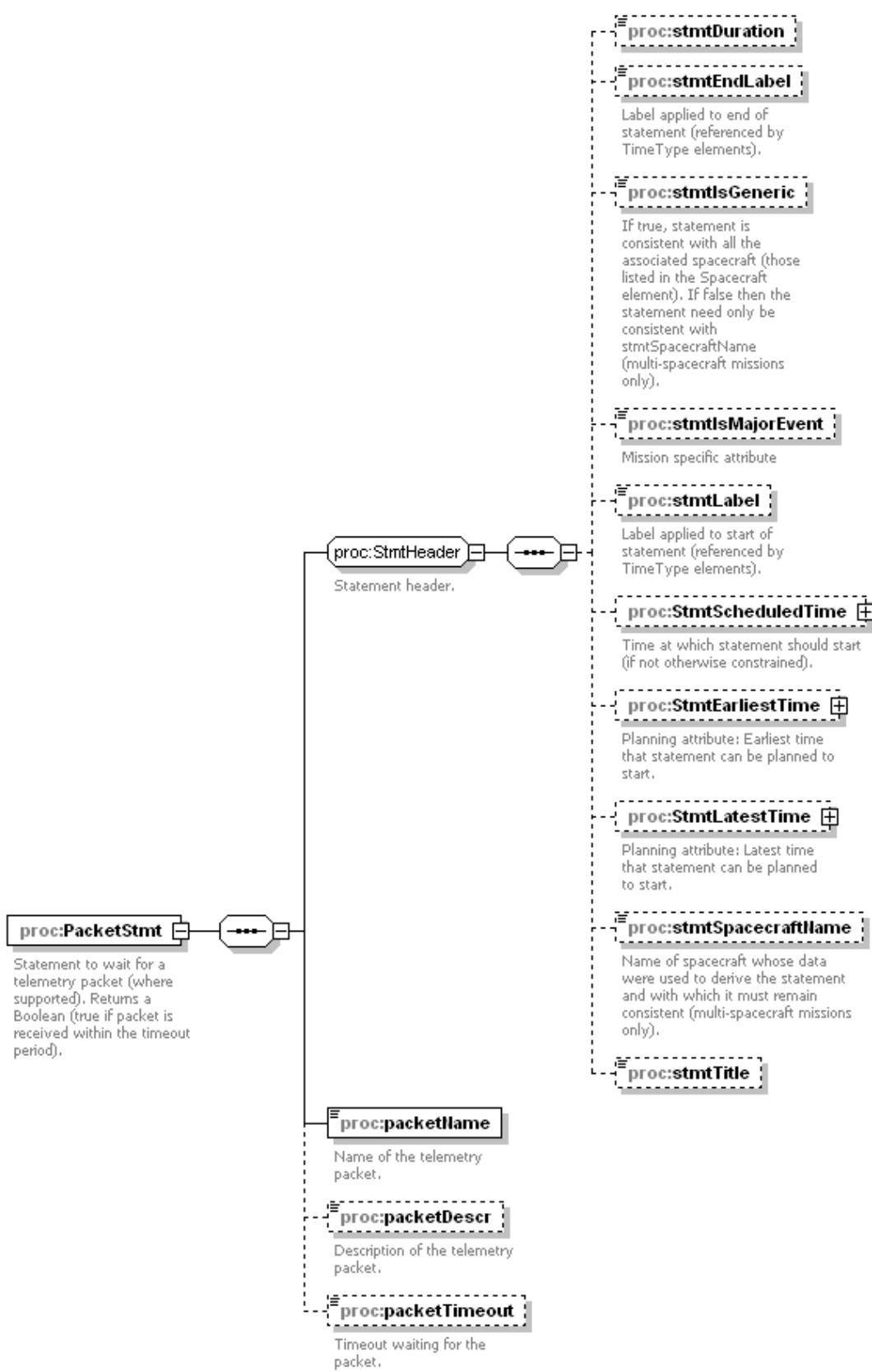
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:checkVariableName proc:CheckVariableValue
annotation	documentation Statement to check a variable against a fixed value, a variable or a telemetry parameter and return a Boolean.
source	<pre><xs:element name="CheckVariableStmt"> <xs:annotation> <xs:documentation>Statement to check a variable against a fixed value, a variable or a telemetry parameter and return a Boolean.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="checkVariableName" type="xs:string"> <xs:annotation> <xs:documentation>Name of Variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> <xs:group ref="proc:CheckVarValue"/> </xs:sequence> </xs:complexType> </xs:element></pre>

element BooleanStmts/CheckVariableStmt/checkVariableName

diagram	 <p>proc:checkVariableName</p> <p>Name of Variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of Variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<pre><xs:element name="checkVariableName" type="xs:string"> <xs:annotation> <xs:documentation>Name of Variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element></pre>

element BooleanStmts/PacketStmt

diagram

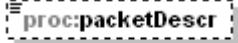


namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:packetName proc:packetDescr proc:packetTimeout
annotation	documentation Statement to wait for a telemetry packet (where supported). Returns a Boolean (true if packet is received within the timeout period).
source	<pre><xs:element name="PacketStmt"> <xs:annotation> <xs:documentation>Statement to wait for a telemetry packet (where supported). Returns a Boolean (true if packet is received within the timeout period).</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:element name="packetName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the telemetry packet.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="packetDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Description of the telemetry packet.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="packetTimeout" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout waiting for the packet.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

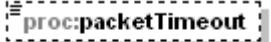
element BooleanStmts/PacketStmt/packetName

diagram	<p>proc:packetName</p> <p>Name of the telemetry packet.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of the telemetry packet.
source	<pre><xs:element name="packetName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the telemetry packet.</xs:documentation> </xs:annotation> </xs:element></pre>

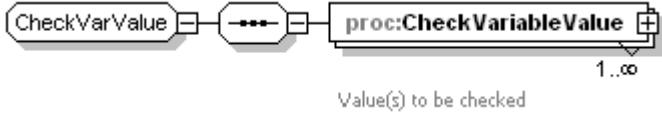
element BooleanStmts/PacketStmt/packetDescr

diagram	 Description of the telemetry packet.								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation Description of the telemetry packet.</p>								
source	<pre><xs:element name="packetDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Description of the telemetry packet.</xs:documentation> </xs:annotation> </xs:element></pre>								

element BooleanStmts/PacketStmt/packetTimeout

diagram	 Timeout waiting for the packet.								
namespace	http://www.omg.org/space/procspec								
type	xs:duration								
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation Timeout waiting for the packet.</p>								
source	<pre><xs:element name="packetTimeout" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout waiting for the packet.</xs:documentation> </xs:annotation> </xs:element></pre>								

group CheckVarValue

diagram	 Value(s) to be checked
namespace	http://www.omg.org/space/procspec
children	proc:CheckVariableValue
used by	elements BooleanStmts/CheckVariableStmt StepType/FORstep/TestFORstmt

source	<pre> <xs:group name="CheckVarValue"> <xs:sequence> <xs:element name="CheckVariableValue" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Value(s) to be checked</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="checkVariableValueId" type="xs:int"> <xs:annotation> <xs:documentation>Unique value check identifier</xs:documentation> </xs:annotation> </xs:element> <xs:element name="checkVariableValueRelationEnum" type="proc:relationEnum"> <xs:annotation> <xs:documentation>Standard relation operators</xs:documentation> </xs:annotation> </xs:element> <xs:element name="checkVariableValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>Type of check - against value, Telemetry or variable</xs:documentation> </xs:annotation> </xs:element> <xs:element name="checkVariableValue" type="xs:string"> <xs:annotation> <xs:documentation>Value to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:group> </pre>
--------	---

element CheckVarValue/CheckVariableValue

diagram	<pre> classDiagram class proc::CheckVariableValue { <<Value(s) to be checked>> } class proc::checkVariableValueId { <<Unique value check identifier>> } class proc::checkVariableValueRelationEnum { <<Standard relation operators>> } class proc::checkVariableValueRadixEnum { <<Type of check - against value, Telemetry or variable>> } proc::CheckVariableValue "1..∞" --> proc::checkVariableValueId proc::CheckVariableValue "1..∞" --> proc::checkVariableValueRelationEnum proc::CheckVariableValue "1..∞" --> proc::checkVariableValueRadixEnum </pre>
---------	--

namespace	http://www.omg.org/space/procspec								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>1</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	1	maxOcc	unbounded	content	complex
isRef	0								
minOcc	1								
maxOcc	unbounded								
content	complex								
children	proc:checkVariableValueId proc:checkVariableValueRelationEnum proc:checkVariableValueRadixEnum proc:checkVariableValue								
annotation	documentation Value(s) to be checked								
source	<pre><xs:element name="CheckVariableValue" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Value(s) to be checked</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="checkVariableValueId" type="xs:int"> <xs:annotation> <xs:documentation>Unique value check identifier</xs:documentation> </xs:annotation> </xs:element> <xs:element name="checkVariableValueRelationEnum" type="proc:relationEnum"> <xs:annotation> <xs:documentation>Standard relation operators</xs:documentation> </xs:annotation> </xs:element> <xs:element name="checkVariableValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>Type of check - against value, Telemetry or variable</xs:documentation> </xs:annotation> </xs:element> <xs:element name="checkVariableValue" type="xs:string"> <xs:annotation> <xs:documentation>Value to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>								

element CheckVarValue/CheckVariableValue/checkVariableValueId

diagram	proc:checkVariableValueId				
	Unique value check identifier				
namespace	http://www.omg.org/space/procspec				
type	xs:int				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
annotation	documentation Unique value check identifier				
source	<pre><xs:element name="checkVariableValueId" type="xs:int"> <xs:annotation></pre>				

	<pre><xs:documentation>Unique value check identifier</xs:documentation> </xs:annotation> </xs:element></pre>
--	--

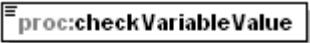
element CheckVarValue/CheckVariableValue/checkVariableValueRelationEnum

diagram	proc:checkVariableValueRelation... Standard relation operators
namespace	http://www.omg.org/space/procspec
type	proc:relationEnum
properties	isRef 0 content simple
facets	enumeration EQUALS enumeration NOTEQUALS enumeration LESSTHAN enumeration LESSTHANOEQUALS enumeration GREATERTHAN enumeration GREATERTHANOEQUALS
annotation	documentation Standard relation operators
source	<pre><xs:element name="checkVariableValueRelationEnum" type="proc:relationEnum"> <xs:annotation> <xs:documentation>Standard relation operators</xs:documentation> </xs:annotation> </xs:element></pre>

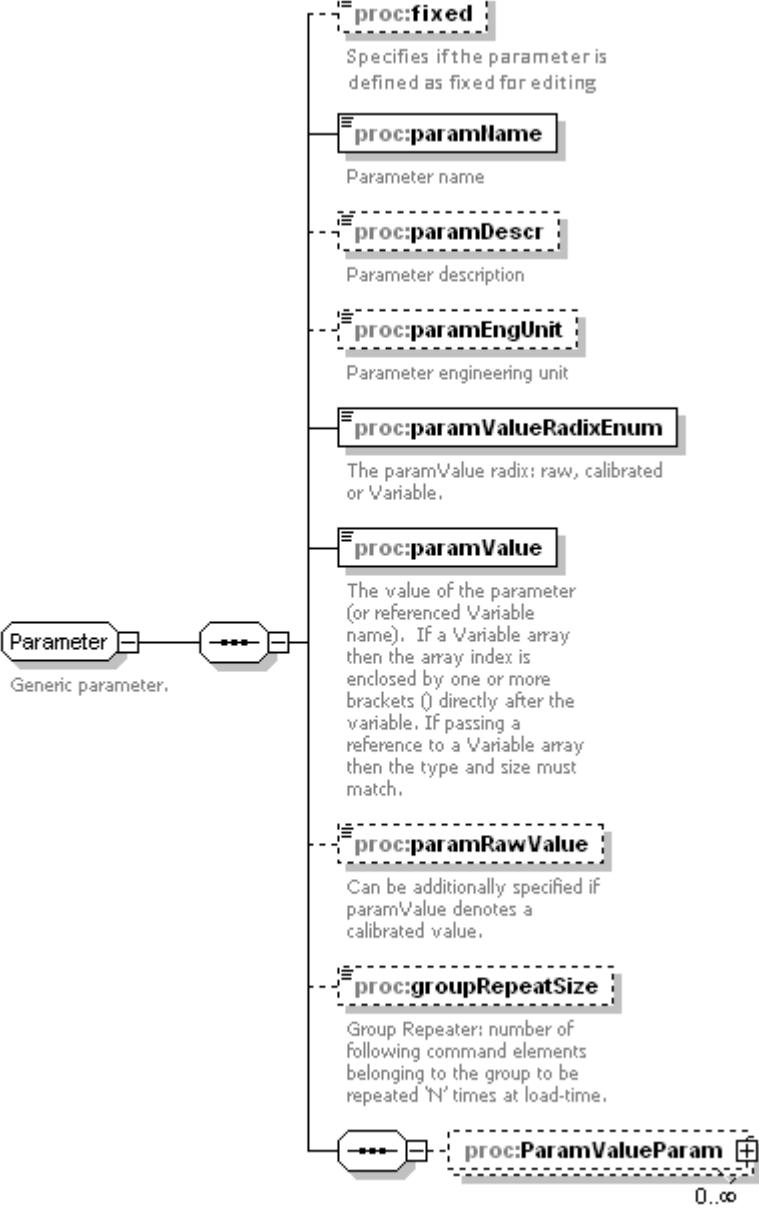
element CheckVarValue/CheckVariableValue/checkVariableValueRadixEnum

diagram	proc:checkVariableValueRadix... Type of check - against value, Telemetry or variable
namespace	http://www.omg.org/space/procspec
type	proc:radixEnum
properties	isRef 0 content simple
annotation	documentation Type of check - against value, Telemetry or variable
source	<pre><xs:element name="checkVariableValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>Type of check - against value, Telemetry or variable</xs:documentation> </xs:annotation> </xs:element></pre>

element **CheckVarValue/CheckVariableValue/checkVariableValue**

diagram	
	Value to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Value to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<pre><xs:element name="checkVariableValue" type="xs:string"> <xs:annotation> <xs:documentation>Value to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element></pre>

group Parameter

diagram	 <p>Parameter Generic parameter.</p> <p>proc:fixed Specifies if the parameter is defined as fixed for editing</p> <p>proc:paramName Parameter name</p> <p>proc:paramDescr Parameter description</p> <p>proc:paramEngUnit Parameter engineering unit</p> <p>proc:paramValueRadixEnum The paramValue radix: raw, calibrated or Variable.</p> <p>proc:paramValue The value of the parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. If passing a reference to a Variable array then the type and size must match.</p> <p>proc:paramRawValue Can be additionally specified if paramValue denotes a calibrated value.</p> <p>proc:groupRepeatSize Group Repeater: number of following command elements belonging to the group to be repeated 'N' times at load-time.</p> <p>proc:ParamValueParam Used for nested parameters, e.g. TC as TC parameter.</p>
namespace	http://www.omg.org/space/procspec
children	proc:paramName proc:paramDescr proc:paramEngUnit proc:paramValueRadixEnum proc:paramValue proc:paramRawValue proc:groupRepeatSize proc:ParamValueParam
used by	elements AllStmts/CmdStmt/CmdParam AllStmts/DirectiveStmt/DirectiveArgument Parameter/ParamValueParam AllStmts/ProcCallStmt/ProcParam AllStmts/SeqCallStmt/SeqParam
annotation	documentation Generic parameter.
source	<pre><xs:group name="Parameter"> <xs:annotation> <xs:documentation>Generic parameter.</xs:documentation> </xs:annotation></pre>

```

<xs:sequence>
  <xs:element name=" fixed " type="xs:boolean" default="false" minOccurs="0">
    <xs:annotation>
      <xs:documentation> Specifies if the parameter is defined as fixed for
editing</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="paramName" type="xs:string">
    <xs:annotation>
      <xs:documentation>Parameter name</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="paramDescr" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Parameter description</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="paramEngUnit" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Parameter engineering unit</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="paramValueRadixEnum" type="proc:radixEnum">
    <xs:annotation>
      <xs:documentation>The paramValue radix: raw, calibrated or Variable.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="paramValue" type="xs:string">
    <xs:annotation>
      <xs:documentation>The value of the parameter (or referenced Variable name). If a Variable
array then the array index is enclosed by one or more brackets () directly after the variable. If
passing a reference to a Variable array then the type and size must match.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="paramRawValue" type="xs:integer" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Can be additionally specified if paramValue denotes a calibrated
value.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="groupRepeatSize" type="xs:integer" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Group Repeater: number of following command elements belonging to
the group to be repeated 'N' times at load-time.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:sequence>
    <xs:element name="ParamValueParam" minOccurs="0" maxOccurs="unbounded">
      <xs:annotation>
        <xs:documentation>Used for nested parameters, e.g. TC as TC
parameter.</xs:documentation>
      </xs:annotation>
      <xs:complexType>
        <xs:group ref="proc:Parameter"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>

```

	<pre></xs:sequence> </xs:sequence> </xs:group></pre>
--	--

element Parameter/fixed

diagram	 <p>proc:fixed</p> <p>Specifies if the parameter is defined as fixed for editing</p>										
namespace	http://www.omg.org/space/procspec										
type	xs:boolean										
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> <tr> <td>default</td> <td>false</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	false
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	false										
annotation	documentation Parameter name										
source	<pre><xs:element name="fixed" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation> Specifies if the parameter is defined as fixed for editing</xs:documentation> </xs:annotation> </xs:element></pre>										

element Parameter/paramName

diagram	 <p>proc:paramName</p> <p>Parameter name</p>				
namespace	http://www.omg.org/space/procspec				
type	xs:string				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
annotation	documentation Parameter name				
source	<pre><xs:element name="paramName" type="xs:string"> <xs:annotation> <xs:documentation>Parameter name</xs:documentation> </xs:annotation> </xs:element></pre>				

element Parameter/paramDescr

diagram	 <p>proc: paramDescr</p> <p>Parameter description</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								

annotation	documentation Parameter description
source	<pre><xs:element name="paramDescr" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Parameter description</xs:documentation> </xs:annotation> </xs:element></pre>

element Parameter/paramEngUnit

diagram	 proc:paramEngUnit Parameter engineering unit								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	documentation Parameter engineering unit								
source	<pre><xs:element name="paramEngUnit" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Parameter engineering unit</xs:documentation> </xs:annotation> </xs:element></pre>								

element Parameter/paramValueRadixEnum

diagram	 proc:paramValueRadixEnum The paramValue radix: raw, calibrated or Variable.				
namespace	http://www.omg.org/space/procspec				
type	proc:radixEnum				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
annotation	documentation The paramValue radix: raw, calibrated or Variable.				
source	<pre><xs:element name="paramValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>The paramValue radix: raw, calibrated or Variable.</xs:documentation> </xs:annotation> </xs:element></pre>				

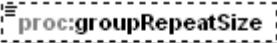
element Parameter/paramValue

diagram	 proc:paramValue
	The value of the parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. If passing a reference to a Variable array then the type and size must match.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The value of the parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. If passing a reference to a Variable array then the type and size must match.
source	<pre><xs:element name="paramValue" type="xs:string"> <xs:annotation> <xs:documentation>The value of the parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. If passing a reference to a Variable array then the type and size must match.</xs:documentation> </xs:annotation> </xs:element></pre>

element Parameter/paramRawValue

diagram	 proc:paramRawValue
	Can be additionally specified if paramValue denotes a calibrated value.
namespace	http://www.omg.org/space/procspec
type	xs:integer
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Can be additionally specified if paramValue denotes a calibrated value.
source	<pre><xs:element name="paramRawValue" type="xs:integer" minOccurs="0"> <xs:annotation> <xs:documentation>Can be additionally specified if paramValue denotes a calibrated value.</xs:documentation> </xs:annotation> </xs:element></pre>

element Parameter/groupRepeatSize

diagram									
	Group Repeater: number of following command elements belonging to the group to be repeated 'N' times at load-time.								
namespace	http://www.omg.org/space/procspec								
type	xs:integer								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>Group Repeater: number of following command elements belonging to the group to be repeated 'N' times at load-time.</p>								
source	<pre><xs:element name="groupRepeatSize" type="xs:integer" minOccurs="0"> <xs:annotation> <xs:documentation>Group Repeater: number of following command elements belonging to the group to be repeated 'N' times at load-time.</xs:documentation> </xs:annotation> </xs:element></pre>								

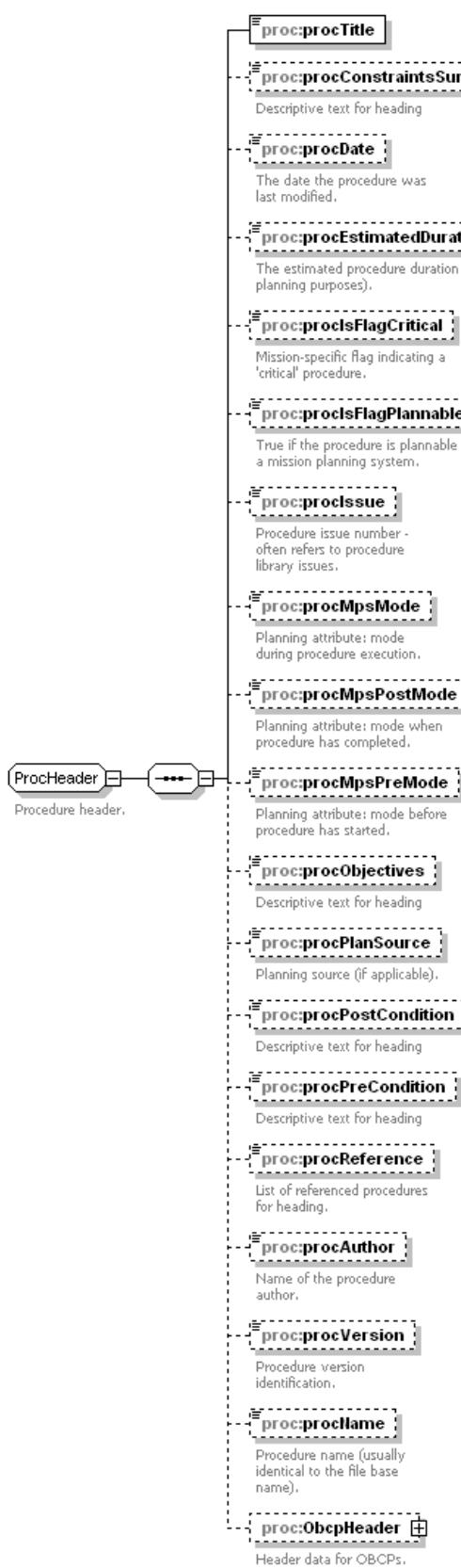
element Parameter/ParamValueParam

diagram	<p>proc:ParamValueParam $0..\infty$</p> <p>Used for nested parameters, e.g. TC as TC parameter.</p> <p>Generic parameter.</p> <p>proc:parameterName Parameter name</p> <p>proc:paramDescr Parameter description</p> <p>proc:paramEngUnit Parameter engineering unit</p> <p>proc:paramValueRadixEnum The paramValue radix: raw, calibrated or Variable.</p> <p>proc:ParameterValue The value of the parameter (or referenced Variable name). If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. If passing a reference to a Variable array then the type and size must match.</p> <p>proc:paramRawValue Can be additionally specified if paramValue denotes a calibrated value.</p> <p>proc:groupRepeatSize Group Repeater: number of following command elements belonging to the group to be repeated 'N' times at load-time.</p> <p>proc:ParamValueParam $0..\infty$</p> <p>Used for nested parameters, e.g. TC as TC parameter.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table border="0"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	complex
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	complex								
children	proc:parameterName proc:paramDescr proc:paramEngUnit proc:paramValueRadixEnum proc:ParameterValue proc:paramRawValue proc:groupRepeatSize proc:ParamValueParam								
annotation	<p>documentation Used for nested parameters, e.g. TC as TC parameter.</p>								
source	<pre><xs:element name="ParamValueParam" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>Used for nested parameters, e.g. TC as TC parameter.</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:Parameter"/> </xs:complexType></pre>								

	</xs:element>
--	---------------

group ProcHeader

diagram



namespace	http://www.omg.org/space/procspec
children	proc:procTitle proc:procConstraintsSummary proc:procDate proc:procEstimatedDuration proc:procIsFlagCritical proc:procIsFlagPlannable proc:procIssue proc:procMpsMode proc:procMpsPostMode proc:procMpsPreMode proc:procObjectives proc:procPlanSource proc:procPostCondition proc:procPreCondition proc:procReference proc:procAuthor proc:procVersion proc:procName proc:ObcpHeader
used by	element Proc/ProcBody
annotation	documentation Procedure header.
source	<pre> <xs:group name="ProcHeader"> <xs:annotation> <xs:documentation>Procedure header.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="procTitle" type="xs:string"/> <xs:element name="procConstraintsSummary" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Descriptive text for heading</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procDate" type="xs:dateTime" minOccurs="0"> <xs:annotation> <xs:documentation>The date the procedure was last modified.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procEstimatedDuration" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>The estimated procedure duration (for planning purposes).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procIsFlagCritical" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>Mission-specific flag indicating a 'critical' procedure.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procIsFlagPlannable" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>True if the procedure is plannable in a mission planning system.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procIssue" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Procedure issue number - often refers to procedure library issues.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procMpsMode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: mode during procedure execution.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="procMpsPostMode" type="xs:string" minOccurs="0"> <xs:annotation> </pre>

```

<xs:documentation>Planning attribute: mode when procedure has completed.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procMpsPreMode" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Planning attribute: mode before procedure has started.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procObjectives" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Descriptive text for heading</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procPlanSource" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Planning source (if applicable).</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procPostCondition" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Descriptive text for heading</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procPreCondition" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Descriptive text for heading</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procReference" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>List of referenced procedures for heading.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procAuthor" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Name of the procedure author.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procVersion" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Procedure version identification.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="procName" type="xs:string" minOccurs="0">
<xs:annotation>
<xs:documentation>Procedure name (usually identical to the file base name).</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="ObcpHeader" minOccurs="0">
<xs:annotation>
<xs:documentation>Header data for OBCPs.</xs:documentation>
</xs:annotation>
</xs:complexType>

```

```

<xs:sequence>
  <xs:element name="obcpId" type="xs:int">
    <xs:annotation>
      <xs:documentation>32-bit integer defining the OBCP.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="obcpName" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Long readable name of the OBCP.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="obcpCode" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Short mnemonic of the OBCP.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="pidName" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Long readable name of the application process.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="pidCode" type="xs:string" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Short mnemonic of the application process.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="pidNumber" type="xs:int" minOccurs="0">
    <xs:annotation>
      <xs:documentation>Numeric identifier of the application process.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="tmPeriod" type="xs:short">
    <xs:annotation>
      <xs:documentation>15-bit unsigned integer defining the periodicity of OBCP TM generation in number of OBSW cycles.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="tmEnabled" type="xs:boolean">
    <xs:annotation>
      <xs:documentation>Flag defining whether OBCP TM generation is activated.</xs:documentation>
    </xs:annotation>
  </xs:element>
  <xs:element name="ObcpParameter" minOccurs="0" maxOccurs="unbounded">
    <xs:annotation>
      <xs:documentation>List of OBCP parameters referenced by the procedure.</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence>
        <xs:element name="obcpPrmName" type="xs:string">
          <xs:annotation>
            <xs:documentation>Name of the OBCP parameter.</xs:documentation>
          </xs:annotation>
        </xs:element>
        <xs:element name="obcpPrmProvided" type="xs:boolean">

```

```

<xs:annotation>
  <xs:documentation>Indicates whether the OBCP parameter is provided by the
procedure, otherwise it is only used.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="obcpPrmReadOnly" type="xs:boolean">
<xs:annotation>
  <xs:documentation>Flag defining whether OBCP parameter is read only or
read/write.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="obcpPrmTypeEnum" type="proc:variableTypeEnum"
minOccurs="0">
<xs:annotation>
  <xs:documentation>The parameter value type needs to be specified for OBCP
parameters provided only.</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TmParameter" minOccurs="0" maxOccurs="unbounded">
<xs:annotation>
  <xs:documentation>List of TM parameters referenced by the
procedure.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="tmPrmName" type="xs:string">
<xs:annotation>
  <xs:documentation>Name of the TM parameter</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Tc" minOccurs="0" maxOccurs="unbounded">
<xs:annotation>
  <xs:documentation>List of telecommands referenced by the
procedure.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="tcName" type="xs:string">
<xs:annotation>
  <xs:documentation>Name of the TC</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Event" minOccurs="0" maxOccurs="unbounded">
<xs:annotation>
  <xs:documentation>List of events referenced by the procedure.</xs:documentation>
</xs:annotation>
</xs:complexType>

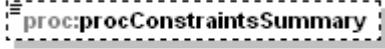
```

	<pre> <xs:sequence> <xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the event</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:group></pre>
--	---

element ProcHeader/procTitle

diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
source	<xs:element name="procTitle" type="xs:string"/>

element ProcHeader/procConstraintsSummary

diagram	
	Descriptive text for heading
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Descriptive text for heading
source	<xs:element name="procConstraintsSummary" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Descriptive text for heading</xs:documentation> </xs:annotation> </xs:element>

element ProcHeader/procDate

diagram	
	The date the procedure was last modified.

namespace	http://www.omg.org/space/procspec
type	xs:dateTime
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The date the procedure was last modified.
source	<pre><xs:element name="procDate" type="xs:dateTime" minOccurs="0"> <xs:annotation> <xs:documentation>The date the procedure was last modified.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procEstimatedDuration

diagram	<p>The estimated procedure duration (for planning purposes).</p>
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation The estimated procedure duration (for planning purposes).
source	<pre><xs:element name="procEstimatedDuration" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>The estimated procedure duration (for planning purposes).</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procIsFlagCritical

diagram	<p>Mission-specific flag indicating a 'critical' procedure.</p>
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Mission-specific flag indicating a 'critical' procedure.
source	<pre><xs:element name="procIsFlagCritical" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>Mission-specific flag indicating a 'critical' procedure.</xs:documentation> </xs:annotation></pre>

	</xs:element>
--	---------------

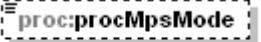
element ProcHeader/procIsFlagPlannable

diagram	 <p>True if the procedure is plannable in a mission planning system.</p>
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation True if the procedure is plannable in a mission planning system.
source	<pre><xs:element name="procIsFlagPlannable" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>True if the procedure is plannable in a mission planning system.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procIssue

diagram	 <p>Procedure issue number - often refers to procedure library issues.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Procedure issue number - often refers to procedure library issues.
source	<pre><xs:element name="procIssue" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Procedure issue number - often refers to procedure library issues.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procMpsMode

diagram	 <p>Planning attribute: mode during procedure execution.</p>
---------	---

namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Planning attribute: mode during procedure execution.
source	<pre><xs:element name="procMpsMode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: mode during procedure execution.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procMpsPostMode

diagram	<p>proc:procMpsPostMode</p> <p>Planning attribute: mode when procedure has completed.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Planning attribute: mode when procedure has completed.
source	<pre><xs:element name="procMpsPostMode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: mode when procedure has completed.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procMpsPreMode

diagram	<p>proc:procMpsPreMode</p> <p>Planning attribute: mode before procedure has started.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Planning attribute: mode before procedure has started.
source	<pre><xs:element name="procMpsPreMode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: mode before procedure has started.</xs:documentation> </xs:annotation></pre>

	</xs:element>
--	---------------

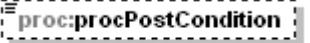
element ProcHeader/procObjectives

diagram	 proc:procObjectives Descriptive text for heading
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Descriptive text for heading
source	<xs:element name="procObjectives" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Descriptive text for heading</xs:documentation> </xs:annotation> </xs:element>

element ProcHeader/procPlanSource

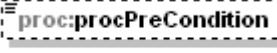
diagram	 proc:procPlanSource Planning source (if applicable).
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Planning source (if applicable).
source	<xs:element name="procPlanSource" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Planning source (if applicable).</xs:documentation> </xs:annotation> </xs:element>

element ProcHeader/procPostCondition

diagram	 proc:procPostCondition Descriptive text for heading
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1

	content simple
annotation	documentation Descriptive text for heading
source	<xs:element name="procPostCondition" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Descriptive text for heading</xs:documentation> </xs:annotation> </xs:element>

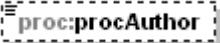
element ProcHeader/procPreCondition

diagram	 Descriptive text for heading
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Descriptive text for heading
source	<xs:element name="procPreCondition" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Descriptive text for heading</xs:documentation> </xs:annotation> </xs:element>

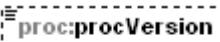
element ProcHeader/procReference

diagram	 List of referenced procedures for heading.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation List of referenced procedures for heading.
source	<xs:element name="procReference" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>List of referenced procedures for heading.</xs:documentation> </xs:annotation> </xs:element>

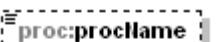
element ProcHeader/procAuthor

diagram	 Name of the procedure author.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Name of the procedure author.
source	<pre><xs:element name="procAuthor" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of the procedure author.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procVersion

diagram	 Procedure version identification.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Procedure version identification.
source	<pre><xs:element name="procVersion" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Procedure version identification.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/procName

diagram	 Procedure name (usually identical to the file base name).
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple

annotation	documentation Procedure name (usually identical to the file base name).
source	<pre><xs:element name="procName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Procedure name (usually identical to the file base name).</xs:documentation> </xs:annotation> </xs:element></pre>

element **ProcHeader/ObcpHeader**

diagram	<pre> classDiagram class proc:obcpId class proc:obcpName class proc:obcpCode class proc:pidName class proc:pidCode class proc:pidNumber class proc:ObcpHeader class proc:tmPeriod class proc:tmEnabled class proc:ObcpParameter class proc:TmParameter class proc:Tc class proc:Event proc:ObcpHeader --> proc:obcpId proc:ObcpHeader --> proc:obcpName proc:ObcpHeader --> proc:obcpCode proc:ObcpHeader --> proc:pidName proc:ObcpHeader --> proc:pidCode proc:ObcpHeader --> proc:pidNumber proc:ObcpHeader --> proc:tmPeriod proc:ObcpHeader --> proc:tmEnabled proc:ObcpHeader --> proc:ObcpParameter proc:ObcpHeader --> proc:TmParameter proc:ObcpHeader --> proc:Tc proc:ObcpHeader --> proc:Event </pre> <p>The diagram illustrates the structure of the ProcHeader/ObcpHeader element. It consists of several attributes and a reference to the ObcpHeader class. The attributes are:</p> <ul style="list-style-type: none"> proc:obcpId: 32-bit integer defining the OBCP. proc:obcpName: Long readable name of the OBCP. proc:obcpCode: Short mnemonic of the OBCP. proc:pidName: Long readable name of the application process. proc:pidCode: Short mnemonic of the application process. proc:pidNumber: Numeric identifier of the application process. proc:tmPeriod: 15-bit unsigned integer defining the periodicity of OBCP TM generation in number of OBSW cycles. proc:tmEnabled: Flag defining whether OBCP TM generation is activated. proc:ObcpParameter: A list of OBCP parameters referenced by the procedure, with a multiplicity of 0..∞. proc:TmParameter: A list of TM parameters referenced by the procedure, with a multiplicity of 0..∞. proc:Tc: A list of telecommands referenced by the procedure, with a multiplicity of 0..∞. proc:Event: A list of events referenced by the procedure, with a multiplicity of 0..∞. <p>The proc:ObcpHeader class is shown with a dashed border and a self-referencing association, indicating it is a composite structure.</p>
namespace	http://www.omg.org/space/procspec

properties	isRef 0 minOcc 0 maxOcc 1 content complex
children	<u>proc:obcpId</u> <u>proc:obcpName</u> <u>proc:obcpCode</u> <u>proc:pidName</u> <u>proc:pidCode</u> <u>proc:pidNumber</u> <u>proc:tmPeriod</u> <u>proc:tmEnabled</u> <u>proc:ObcmParameter</u> <u>proc:TmParameter</u> <u>proc:Tc</u> <u>proc:Event</u>
annotation	documentation Header data for OBCPs.
source	<pre><xs:element name="ObcpHeader" minOccurs="0"> <xs:annotation> <xs:documentation>Header data for OBCPs.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="obcpId" type="xs:int"> <xs:annotation> <xs:documentation>32-bit integer defining the OBCP.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Long readable name of the OBCP.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpCode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Short mnemonic of the OBCP.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="pidName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Long readable name of the application process.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="pidCode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Short mnemonic of the application process.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="pidNumber" type="xs:int" minOccurs="0"> <xs:annotation> <xs:documentation>Numeric identifier of the application process.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tmPeriod" type="xs:short"> <xs:annotation> <xs:documentation>15-bit unsigned integer defining the periodicity of OBCP TM generation in number of OBSW cycles.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="tmEnabled" type="xs:boolean"> <xs:annotation> <xs:documentation>Flag defining whether OBCP TM generation is activated.</xs:documentation> </xs:annotation> </xs:element></pre>

	<pre> <xs:element name="ObcpParameter" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>List of OBCP parameters referenced by the procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="obcpPrmName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the OBCP parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpPrmProvided" type="xs:boolean"> <xs:annotation> <xs:documentation>Indicates whether the OBCP parameter is provided by the procedure, otherwise it is only used.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpPrmReadOnly" type="xs:boolean"> <xs:annotation> <xs:documentation>Flag defining whether OBCP parameter is read only or read/write.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpPrmTypeEnum" type="proc:variableTypeEnum" minOccurs="0"> <xs:annotation> <xs:documentation>The parameter value type needs to be specified for OBCP parameters provided only.</xs:documentation> </xs:annotation> </xs:element> <xs:sequence> <xs:complexType> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="TmParameter" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>List of TM parameters referenced by the procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="tmPrmName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the TM parameter</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="Tc" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>List of telecommands referenced by the procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="tcName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the TC</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

```

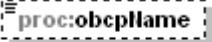
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Event" minOccurs="0" maxOccurs="unbounded">
<xs:annotation>
<xs:documentation>List of events referenced by the procedure.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="eventName" type="xs:string">
<xs:annotation>
<xs:documentation>Name of the event</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>

```

element ProcHeader/ObcpHeader/obcpId

diagram	 proc:obcpId 32-bit integer defining the OBCP.
namespace	http://www.omg.org/space/procspec
type	xs:int
properties	isRef 0 content simple
annotation	documentation 32-bit integer defining the OBCP.
source	<xs:element name="obcpId" type="xs:int"> <xs:annotation> <xs:documentation>32-bit integer defining the OBCP.</xs:documentation> </xs:annotation> </xs:element>

element ProcHeader/ObcpHeader/obcpName

diagram	 proc:obcpName Long readable name of the OBCP.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1

	content simple
annotation	documentation Long readable name of the OBCP.
source	<pre><xs:element name="obcpName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Long readable name of the OBCP.</xs:documentation> </xs:annotation> </xs:element></pre>

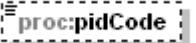
element ProcHeader/ObcmHeader/obcpCode

diagram	 Short mnemonic of the OBCP.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Short mnemonic of the OBCP.
source	<pre><xs:element name="obcpCode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Short mnemonic of the OBCP.</xs:documentation> </xs:annotation> </xs:element></pre>

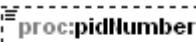
element ProcHeader/ObcmHeader/pidName

diagram	 Long readable name of the application process.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Long readable name of the application process.
source	<pre><xs:element name="pidName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Long readable name of the application process.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcmHeader/pidCode

diagram	 Short mnemonic of the application process.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Short mnemonic of the application process.
source	<pre><xs:element name="pidCode" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Short mnemonic of the application process.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcmHeader/pidNumber

diagram	 Numeric identifier of the application process.
namespace	http://www.omg.org/space/procspec
type	xs:int
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Numeric identifier of the application process.
source	<pre><xs:element name="pidNumber" type="xs:int" minOccurs="0"> <xs:annotation> <xs:documentation>Numeric identifier of the application process.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcmHeader/tmPeriod

diagram	 15-bit unsigned integer defining the periodicity of OBCP TM generation in number of OBSW cycles.
namespace	http://www.omg.org/space/procspec
type	xs:short
properties	isRef 0 content simple
annotation	documentation 15-bit unsigned integer defining the periodicity of OBCP TM generation in number of OBSW cycles.

source	<pre><xs:element name="tmPeriod" type="xs:short"> <xs:annotation> <xs:documentation>15-bit unsigned integer defining the periodicity of OBCP TM generation in number of OBSW cycles.</xs:documentation> </xs:annotation> </xs:element></pre>
--------	--

element ProcHeader/ObcpHeader/tmEnabled

diagram	<p>Flag defining whether OBCP TM generation is activated.</p>
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 content simple
annotation	documentation Flag defining whether OBCP TM generation is activated.
source	<pre><xs:element name="tmEnabled" type="xs:boolean"> <xs:annotation> <xs:documentation>Flag defining whether OBCP TM generation is activated.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcpHeader/ObcpParameter

diagram	<p>List of OBCP parameters referenced by the procedure.</p> <p>proc:obcpPrmName Name of the OBCP parameter.</p> <p>proc:obcpPrmProvided Indicates whether the OBCP parameter is provided by the procedure, otherwise it is only used.</p> <p>proc:obcpPrmReadOnly Flag defining whether OBCP parameter is read only or read/write.</p> <p>proc:obcpPrmTypeEnum The parameter value type needs to be specified for OBCP parameters provided only.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc unbounded content complex

children	proc:obcpPrmName proc:obcpPrmProvided proc:obcpPrmReadOnly proc:obcpPrmTypeEnum
annotation	documentation List of OBCP parameters referenced by the procedure.
source	<pre><xs:element name="ObcpParameter" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>List of OBCP parameters referenced by the procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="obcpPrmName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the OBCP parameter.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpPrmProvided" type="xs:boolean"> <xs:annotation> <xs:documentation>Indicates whether the OBCP parameter is provided by the procedure, otherwise it is only used.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpPrmReadOnly" type="xs:boolean"> <xs:annotation> <xs:documentation>Flag defining whether OBCP parameter is read only or read/write.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="obcpPrmTypeEnum" type="proc:variableTypeEnum" minOccurs="0"> <xs:annotation> <xs:documentation>The parameter value type needs to be specified for OBCP parameters provided only.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element ProcHeader/ObcpHeader/ObcpParameter/obcpPrmName

diagram	 <p>Name of the OBCP parameter.</p>				
namespace	http://www.omg.org/space/procspec				
type	xs:string				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
annotation	documentation Name of the OBCP parameter.				
source	<pre><xs:element name="obcpPrmName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the OBCP parameter.</xs:documentation> </xs:annotation> </xs:element></pre>				

element ProcHeader/ObcmHeader/ObcmParameter/obcmPrmProvided

diagram	 proc:obcmPrmProvided
	Indicates whether the OBCP parameter is provided by the procedure, otherwise it is only used.
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 content simple
annotation	documentation Indicates whether the OBCP parameter is provided by the procedure, otherwise it is only used.
source	<pre><xs:element name="obcmPrmProvided" type="xs:boolean"> <xs:annotation> <xs:documentation>Indicates whether the OBCP parameter is provided by the procedure, otherwise it is only used.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcmHeader/ObcmParameter/obcmPrmReadOnly

diagram	 proc:obcmPrmReadOnly
	Flag defining whether OBCP parameter is read only or read/write.
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 content simple
annotation	documentation Flag defining whether OBCP parameter is read only or read/write.
source	<pre><xs:element name="obcmPrmReadOnly" type="xs:boolean"> <xs:annotation> <xs:documentation>Flag defining whether OBCP parameter is read only or read/write.</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcmHeader/ObcmParameter/obcmPrmTypeEnum

diagram	 proc:obcmPrmTypeEnum
	The parameter value type needs to be specified for OBCP parameters provided only.
namespace	http://www.omg.org/space/procspec
type	proc:variableTypeEnum

properties	isRef 0 minOcc 0 maxOcc 1 content simple
facets	enumeration DERIVED enumeration BOOLEAN enumeration BYTE enumeration INTEGER enumeration LONG_INTEGER enumeration UNSIGNED_BYTE enumeration UNSIGNED_INTEGER enumeration UNSIGNED_LONG_INTEGER enumeration REAL enumeration DOUBLE enumeration RELATIVETIME enumeration ABSOLUTETIME enumeration STRING
annotation	documentation The parameter value type needs to be specified for OBCP parameters provided only.
source	<pre><xs:element name="obcpPrmTypeEnum" type="proc:variableTypeEnum" minOccurs="0"> <xs:annotation> <xs:documentation>The parameter value type needs to be specified for OBCP parameters provided only.</xs:documentation> </xs:annotation> </xs:element></pre>

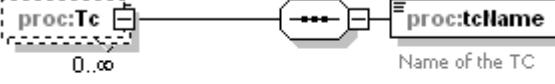
element ProcHeader/ObcpHeader/TmParameter

diagram	<p>The diagram shows a UML class named proc:TmParameter represented by a dashed-line rectangle. It has a directed association to another class named proc:tmPrmName, also in a dashed-line rectangle. The multiplicity at the proc:TmParameter end is 0..oo. A label below the diagram states: "List of TM parameters referenced by the procedure."</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc unbounded content complex
children	proc:tmPrmName
annotation	documentation List of TM parameters referenced by the procedure.
source	<pre><xs:element name="TmParameter" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>List of TM parameters referenced by the procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="tmPrmName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the TM parameter</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element ProcHeader/ObcmHeader/TmParameter/tmPrmName

diagram	 Name of the TM parameter
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc unbounded content simple
annotation	documentation Name of the TM parameter
source	<pre><xs:element name="tmPrmName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the TM parameter</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcmHeader/Tc

diagram	 Name of the TC List of telecommands referenced by the procedure.
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc unbounded content complex
children	proc:tcName
annotation	documentation List of telecommands referenced by the procedure.
source	<pre><xs:element name="Tc" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>List of telecommands referenced by the procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="tcName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the TC</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element ProcHeader/ObcmHeader/Tc/tcName

diagram	 Name of the TC
---------	--

namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of the TC
source	<pre><xs:element name="tcName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the TC</xs:documentation> </xs:annotation> </xs:element></pre>

element ProcHeader/ObcmHeader/Event

diagram	<p>proc:Event</p> <p>0..oo</p> <p>List of events referenced by the procedure.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc unbounded content complex
children	proc:eventName
annotation	documentation List of events referenced by the procedure.
source	<pre><xs:element name="Event" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>List of events referenced by the procedure.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the event</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element ProcHeader/ObcmHeader/Event(eventName)

diagram	<p>proc:eventName</p> <p>Name of the event</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple

annotation	documentation Name of the event
source	<pre><xs:element name="eventName" type="xs:string"> <xs:annotation> <xs:documentation>Name of the event</xs:documentation> </xs:annotation> </xs:element></pre>

group SeqHeader

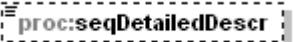
diagram	<p>Sequence header.</p>
namespace	http://www.omg.org/space/procspec
children	proc:seqDescr proc:seqDetailedDescr proc:seqFlagCritical proc:seqFlagPlannable proc:seqStandAlone proc:seqSubSys proc:seqSubSchedId proc:seqGroupId proc:seqFlagSchedule
used by	element Proc/Sequence/SeqHeader
annotation	documentation Sequence header.
source	<pre><xs:group name="SeqHeader"> <xs:annotation> <xs:documentation>Sequence header.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="seqDescr" type="xs:string"/> <xs:element name="seqDetailedDescr" type="xs:string" minOccurs="0"/> <xs:element name="seqFlagCritical" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>Mission-specific flag indicating a 'critical' sequence</xs:documentation> </xs:annotation> </xs:element> <xs:element name="seqFlagPlannable" type="xs:string" default="N" minOccurs="0"> <xs:annotation> <xs:documentation>True if the sequence is plannable in a mission planning system.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="seqStandAlone" type="xs:boolean" default="true" minOccurs="0"/> </xs:sequence> </xs:group></pre>

	<pre><xs:element name="seqSubSys" type="xs:int" minOccurs="0"/> <xs:element name="seqSubSchedId" type="xs:int" minOccurs="0"/> </xs:sequence> </xs:group></pre>
--	---

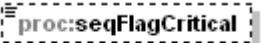
element SeqHeader/seqDescr

diagram					
namespace	http://www.omg.org/space/procspec				
type	xs:string				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				
source	<pre><xs:element name="seqDescr" type="xs:string"/></pre>				

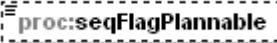
element SeqHeader/seqDetailedDescr

diagram									
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
source	<pre><xs:element name="seqDetailedDescr" type="xs:string" minOccurs="0"/></pre>								

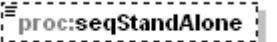
element SeqHeader/seqFlagCritical

diagram											
	Mission-specific flag indicating a 'critical' sequence										
namespace	http://www.omg.org/space/procspec										
type	xs:boolean										
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> <tr> <td>default</td> <td>false</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	false
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	false										
annotation	documentation Mission-specific flag indicating a 'critical' sequence										
source	<pre><xs:element name="seqFlagCritical" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>Mission-specific flag indicating a 'critical' sequence</xs:documentation> </xs:annotation> </xs:element></pre>										

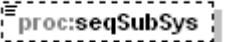
element SeqHeader/seqFlagPlannable

diagram	 proc:seqFlagPlannable True if the sequence is plannable in a mission planning system.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple default N
annotation	documentation True if the sequence is plannable in a mission planning system.
source	<xs:element name="seqFlagPlannable" type="xs:string" default="N" minOccurs="0"> <xs:annotation> <xs:documentation>True if the sequence is plannable in a mission planning system.</xs:documentation> </xs:annotation> </xs:element>

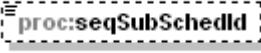
element SeqHeader/seqStandAlone

diagram	 proc:seqStandAlone
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple default true
source	<xs:element name="seqStandAlone" type="xs:boolean" default="true" minOccurs="0"/>

element SeqHeader/seqSubSys

diagram	 proc:seqSubSys
namespace	http://www.omg.org/space/procspec
type	xs:int
properties	isRef 0 minOcc 0 maxOcc 1 content simple
source	<xs:element name="seqSubSys" type="xs:int" minOccurs="0"/>

element SeqHeader/seqSubSchedId

diagram									
namespace	http://www.omg.org/space/procspec								
type	xs:int								
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
source	<xs:element name="seqSubSchedId" type="xs:int" minOccurs="0"/>								

element SeqHeader/seqGroupId

diagram	Proc:seqGroupId								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
source	<xs:element name="seqGroupId" type="xs:string" minOccurs="0"/>								

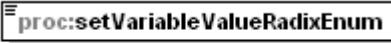
element SeqHeader/seqFlagSchedule

diagram	Proc:seqFlagSchedule										
namespace	http://www.omg.org/space/procspec										
type	xs:string										
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> <tr><td>default</td><td>N</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	N
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	N										
source	<xs:element name="seqFlagSchedule" type="xs:string" default="N" minOccurs="0"/>										

group SetVarValue

diagram	
namespace	http://www.omg.org/space/procspec
children	proc:setVariableValueRadixEnum proc:setVariableValue proc:setVariableExpression proc:setVariableIsUserInput
used by	elements StepType/FORstep/InitFORStmt AllStmts/SetVariableStmt
source	<pre> <xs:group name="SetVarValue"> <xs:sequence> <xs:element name="setVariableValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>Radix of set statement value </xs:documentation> </xs:annotation> </xs:element> <xs:choice> <xs:element name="setVariableValue" type="xs:string"> <xs:annotation> <xs:documentation>Fixed value on RHS (radix not Variable)</xs:documentation> </xs:annotation> </xs:element> <xs:element name="setVariableExpression"> <xs:annotation> <xs:documentation>Arithmetic expression of variables on RHS (radix = Variable)</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element> </xs:choice> <xs:element name="setVariableIsUserInput" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If true procedure execution halts and the user is prompted for the value.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:group></pre>

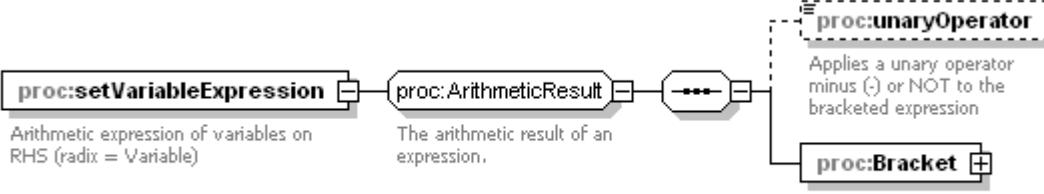
element SetVarValue/setVariableValueRadixEnum

diagram	
	Radix of set statement value
namespace	http://www.omg.org/space/procspec
type	proc:radixEnum
properties	isRef 0 content simple
annotation	documentation Radix of set statement value
source	<pre><xs:element name="setVariableValueRadixEnum" type="proc:radixEnum"> <xs:annotation> <xs:documentation>Radix of set statement value </xs:documentation> </xs:annotation> </xs:element></pre>

element SetVarValue/setVariableValue

diagram	
	Fixed value on RHS (radix not Variable)
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Fixed value on RHS (radix not Variable)
source	<pre><xs:element name="setVariableValue" type="xs:string"> <xs:annotation> <xs:documentation>Fixed value on RHS (radix not Variable)</xs:documentation> </xs:annotation> </xs:element></pre>

element SetVarValue/setVariableExpression

diagram	 <p>The diagram illustrates the structure of a setVariableExpression. It starts with a box labeled 'proc:setVariableExpression' with a minus sign (-) icon. An arrow points from it to a box labeled 'proc:ArithmeticResult'. Another arrow points from 'proc:ArithmeticResult' to three small gray boxes arranged horizontally. From the second of these three boxes, an arrow points to a box labeled 'proc:Bracket' with a plus sign (+) icon. Finally, an arrow points from 'proc:Bracket' to a box labeled 'proc:unaryOperator' with a minus sign (-) icon.</p>
	Arithmetic expression of variables on RHS (radix = Variable)
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:unaryOperator proc:Bracket
annotation	documentation Arithmetic expression of variables on RHS (radix = Variable)

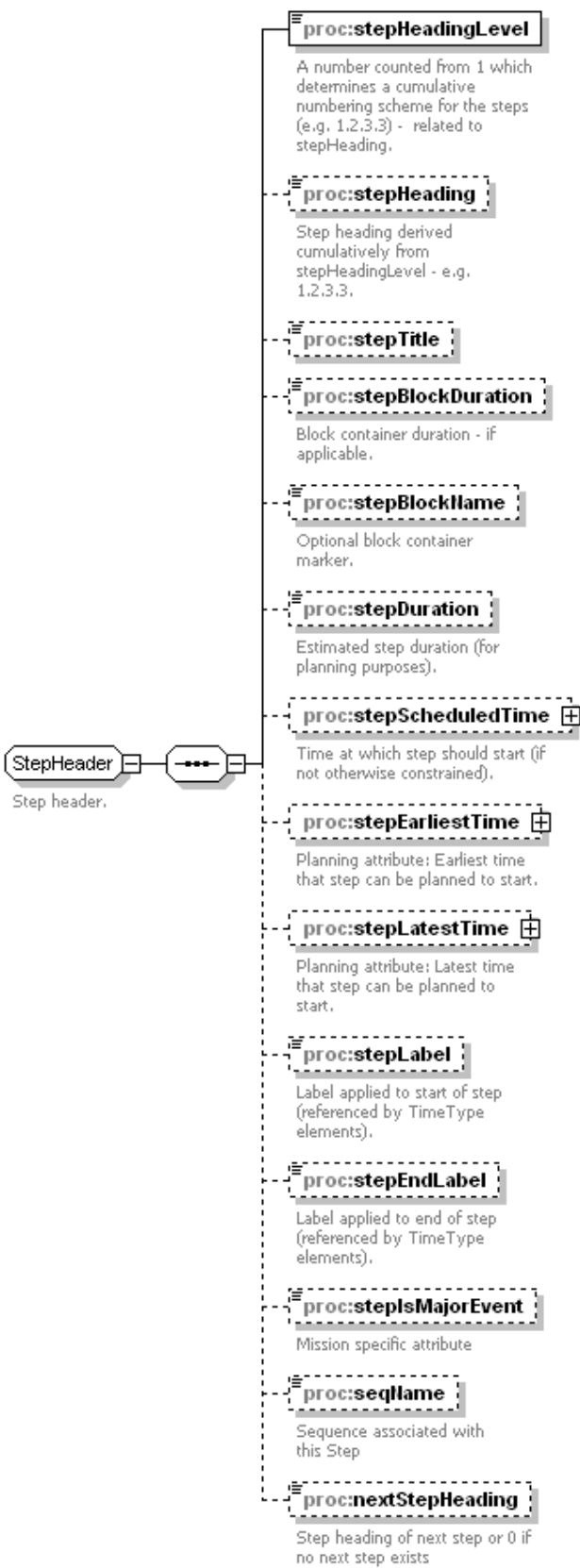
source	<pre><xs:element name="setVariableExpression"> <xs:annotation> <xs:documentation>Arithmetic expression of variables on RHS (radix = Variable)</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element></pre>
--------	--

element SetVarValue/setVariablesUserInput

diagram	 <p>If true procedure execution halts and the user is prompted for the value.</p>										
namespace	http://www.omg.org/space/procspec										
type	xs:boolean										
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> <tr> <td>default</td> <td>false</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	false
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	false										
annotation	<p>documentation If true procedure execution halts and the user is prompted for the value.</p>										
source	<pre><xs:element name="setVariablesUserInput" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If true procedure execution halts and the user is prompted for the value.</xs:documentation> </xs:annotation> </xs:element></pre>										

group StepHeader

diagram



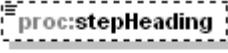
namespace	http://www.omg.org/space/procspec
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepsMajorEvent proc:seqName proc:nextStepHeading
used by	complexType StepType
annotation	documentation Step header.
source	<pre><xs:group name="StepHeader"> <xs:annotation> <xs:documentation>Step header.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="stepHeadingLevel" type="xs:nonNegativeInteger"> <xs:annotation> <xs:documentation>A number counted from 1 which determines a cumulative numbering scheme for the steps (e.g. 1.2.3.3) - related to stepHeading.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepHeading" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Step heading derived cumulatively from stepHeadingLevel - e.g. 1.2.3.3.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepTitle" type="xs:string" minOccurs="0"/> <xs:element name="stepBlockDuration" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Block container duration - if applicable.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepBlockName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Optional block container marker. </xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepDuration" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Estimated step duration (for planning purposes).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepScheduledTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Time at which step should start (if not otherwise constrained).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepEarliestTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: Earliest time that step can be planned to start.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepLatestTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: Latest time that step can be planned to </pre>

	<pre> start.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to start of step (referenced by TimeType elements).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepEndLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to end of step (referenced by TimeType elements).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stepIsMajorEvent" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>Mission specific attribute</xs:documentation> </xs:annotation> </xs:element> <xs:element name="seqName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Sequence associated with this Step</xs:documentation> </xs:annotation> </xs:element> <xs:element name="nextStepHeading" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Step heading of next step or 0 if no next step exists</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:group> </pre>
--	--

element StepHeader/stepHeadingLevel

diagram	 proc:stepHeadingLevel <p>A number counted from 1 which determines a cumulative numbering scheme for the steps (e.g. 1.2.3.3) - related to stepHeading.</p>
namespace	http://www.omg.org/space/procspec
type	xs:nonNegativeInteger
properties	isRef 0 content simple
annotation	<p>documentation</p> <p>A number counted from 1 which determines a cumulative numbering scheme for the steps (e.g. 1.2.3.3) - related to stepHeading.</p>
source	<pre> <xs:element name="stepHeadingLevel" type="xs:nonNegativeInteger"> <xs:annotation> <xs:documentation>A number counted from 1 which determines a cumulative numbering scheme for the steps (e.g. 1.2.3.3) - related to stepHeading.</xs:documentation> </xs:annotation> </xs:element> </pre>

element StepHeader/stepHeading

diagram	 proc:stepHeading Step heading derived cumulatively from stepHeadingLevel - e.g. 1.2.3.3.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Step heading derived cumulatively from stepHeadingLevel - e.g. 1.2.3.3.
source	<xs:element name="stepHeading" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Step heading derived cumulatively from stepHeadingLevel - e.g. 1.2.3.3.</xs:documentation> </xs:annotation> </xs:element>

element StepHeader/stepTitle

diagram	 proc:stepTitle
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
source	<xs:element name="stepTitle" type="xs:string" minOccurs="0"/>

element StepHeader/stepBlockDuration

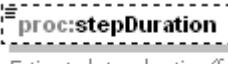
diagram	 proc:stepBlockDuration Block container duration - if applicable.
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Block container duration - if applicable.
source	<xs:element name="stepBlockDuration" type="xs:duration" minOccurs="0">

	<pre><xs:annotation> <xs:documentation>Block container duration - if applicable.</xs:documentation> </xs:annotation> </xs:element></pre>
--	--

element StepHeader/stepBlockName

diagram	 <p>Optional block container marker.</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Optional block container marker.
source	<pre><xs:element name="stepBlockName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Optional block container marker. </xs:documentation> </xs:annotation> </xs:element></pre>

element StepHeader/stepDuration

diagram	 <p>Estimated step duration (for planning purposes).</p>
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Estimated step duration (for planning purposes).
source	<pre><xs:element name="stepDuration" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Estimated step duration (for planning purposes).</xs:documentation> </xs:annotation> </xs:element></pre>

element StepHeader/stepScheduledTime

diagram	<pre> classDiagram class proc:TimeType { proc:stepScheduledTime proc:absoluteTime proc:absoluteTimeLabel proc:negativeSign proc:relativeTime } proc:stepScheduledTime --> proc:TimeType proc:TimeType < -- proc:absoluteTime proc:TimeType < -- proc:absoluteTimeLabel proc:TimeType < -- proc:negativeSign proc:TimeType < -- proc:relativeTime </pre> <p>proc:TimeType</p> <ul style="list-style-type: none"> proc:stepScheduledTime Time at which step should start (if not otherwise constrained). proc:absoluteTime An absolute time literal proc:absoluteTimeLabel A label defined in a Step or Statement proc:absoluteTimeVariable A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. proc:negativeSign proc:relativeTime A relative time literal proc:relativeTimeVariable A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. 								
namespace	http://www.omg.org/space/procspec								
type	proc:TimeType								
properties	<table border="0"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable								
annotation	<p>documentation</p> <p>Time at which step should start (if not otherwise constrained).</p>								
source	<pre> <xs:element name="stepScheduledTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Time at which step should start (if not otherwise constrained).</xs:documentation> </xs:annotation> </xs:element> </pre>								

element StepHeader/stepEarliestTime

diagram	<pre> classDiagram class proc:TimeType { <<proc:TimeType>> <<proc:stepEarliestTime>> <<proc:absoluteTime>> <<proc:absoluteTimeLabel>> <<proc:absoluteTimeVariable>> <<proc:negativeSign>> <<proc:relativeTime>> <<proc:relativeTimeVariable>> } proc:TimeType "1" *-- "1" proc:stepEarliestTime : <<proc:stepEarliestTime>> proc:TimeType "1" *-- "1" proc:absoluteTime : <<proc:absoluteTime>> proc:TimeType "1" *-- "1" proc:absoluteTimeLabel : <<proc:absoluteTimeLabel>> proc:TimeType "1" *-- "1" proc:absoluteTimeVariable : <<proc:absoluteTimeVariable>> proc:TimeType "1" *-- "1" proc:negativeSign : <<proc:negativeSign>> proc:TimeType "1" *-- "1" proc:relativeTime : <<proc:relativeTime>> proc:TimeType "1" *-- "1" proc:relativeTimeVariable : <<proc:relativeTimeVariable>> </pre> <p>The diagram illustrates the structure of the <code>proc:TimeType</code> element. It contains a dashed box labeled <code>proc:stepEarliestTime</code> with the note: "Planning attribute: Earliest time that step can be planned to start." A solid box labeled <code>proc:TimeType</code> contains several associations: <code>proc:absoluteTime</code>, <code>proc:absoluteTimeLabel</code>, <code>proc:absoluteTimeVariable</code>, <code>proc:negativeSign</code>, <code>proc:relativeTime</code>, and <code>proc:relativeTimeVariable</code>. Each association is accompanied by a brief description of its meaning.</p>								
namespace	http://www.omg.org/space/procspec								
type	proc:TimeType								
properties	<table border="1"> <tr> <td data-bbox="187 1129 269 1151">isRef</td><td data-bbox="269 1129 301 1151">0</td></tr> <tr> <td data-bbox="187 1151 269 1172">minOcc</td><td data-bbox="269 1151 301 1172">0</td></tr> <tr> <td data-bbox="187 1172 269 1193">maxOcc</td><td data-bbox="269 1172 301 1193">1</td></tr> <tr> <td data-bbox="187 1193 269 1214">content</td><td data-bbox="269 1193 481 1214">complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable								
annotation	<p>documentation</p> <p>Planning attribute: Earliest time that step can be planned to start.</p>								
source	<pre> <xsd:element name="stepEarliestTime" type="proc:TimeType" minOccurs="0"> <xsd:annotation> <xsd:documentation>Planning attribute: Earliest time that step can be planned to start.</xsd:documentation> </xsd:annotation> </xsd:element> </pre>								

element StepHeader/stepLatestTime

diagram	<pre> classDiagram class proc:TimeType { <<proc:TimeType>> <<proc:stepLatestTime>> <<proc:absoluteTime>> <<proc:absoluteTimeLabel>> <<proc:absoluteTimeVariable>> <<proc:negativeSign>> <<proc:relativeTime>> <<proc:relativeTimeVariable>> } proc:TimeType "1" *-- "1" proc:stepLatestTime proc:TimeType "1" *-- "1" proc:absoluteTime proc:TimeType "1" *-- "1" proc:absoluteTimeLabel proc:TimeType "1" *-- "1" proc:absoluteTimeVariable proc:TimeType "1" *-- "1" proc:negativeSign proc:TimeType "1" *-- "1" proc:relativeTime proc:TimeType "1" *-- "1" proc:relativeTimeVariable </pre> <p>The diagram illustrates the structure of the <code>proc:TimeType</code> element. It is a complex type with several child elements: <code>proc:stepLatestTime</code>, <code>proc:absoluteTime</code>, <code>proc:absoluteTimeLabel</code>, <code>proc:absoluteTimeVariable</code>, <code>proc:negativeSign</code>, <code>proc:relativeTime</code>, and <code>proc:relativeTimeVariable</code>. <code>proc:stepLatestTime</code> is described as a planning attribute representing the latest time a step can be planned to start. <code>proc:absoluteTime</code>, <code>proc:absoluteTimeLabel</code>, and <code>proc:absoluteTimeVariable</code> represent absolute time literals or variables. <code>proc:negativeSign</code> is a sign indicator. <code>proc:relativeTime</code> and <code>proc:relativeTimeVariable</code> represent relative time literals or variables.</p>								
namespace	http://www.omg.org/space/procspec								
type	proc:TimeType								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable								
annotation	<p>documentation</p> <p>Planning attribute: Latest time that step can be planned to start.</p>								
source	<pre> <xsd:element name="stepLatestTime" type="proc:TimeType" minOccurs="0"> <xsd:annotation> <xsd:documentation>Planning attribute: Latest time that step can be planned to start.</xsd:documentation> </xsd:annotation> </xsd:element> </pre>								

element StepHeader/stepLabel

diagram	<pre> classDiagram class proc:stepLabel { <<proc:stepLabel>> } proc:stepLabel *-- "1" proc:TimeType </pre> <p>The diagram shows the <code>proc:stepLabel</code> element as a simple type that is associated with the <code>proc:TimeType</code> element, indicating it is applied to the start of a step.</p>
namespace	http://www.omg.org/space/procspec
type	xsd:string

properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Label applied to start of step (referenced by TimeType elements).
source	<pre><xs:element name="stepLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to start of step (referenced by TimeType elements).</xs:documentation> </xs:annotation> </xs:element></pre>

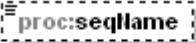
element StepHeader/stepEndLabel

diagram	<p>proc:stepEndLabel</p> <p>Label applied to end of step (referenced by TimeType elements).</p>
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Label applied to end of step (referenced by TimeType elements).
source	<pre><xs:element name="stepEndLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to end of step (referenced by TimeType elements).</xs:documentation> </xs:annotation> </xs:element></pre>

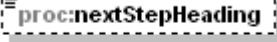
element StepHeader/stepIsMajorEvent

diagram	<p>proc:stepIsMajorEvent</p> <p>Mission specific attribute</p>
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Mission specific attribute
source	<pre><xs:element name="stepIsMajorEvent" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>Mission specific attribute</xs:documentation> </xs:annotation> </xs:element></pre>

element StepHeader/seqName

diagram	 Sequence associated with this Step								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	documentation Sequence associated with this Step								
source	<pre><xs:element name="seqName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Sequence associated with this Step</xs:documentation> </xs:annotation> </xs:element></pre>								

element StepHeader/nextStepHeading

diagram	 Step heading of next step or 0 if no next step exists								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	documentation Step heading of next step or 0 if no next step exists								
source	<pre><xs:element name="nextStepHeading" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Step heading of next step or 0 if no next step exists</xs:documentation> </xs:annotation> </xs:element></pre>								

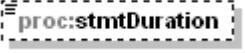
group StmtHeader

diagram	 <p>Statement header.</p> <p>The diagram shows a rounded rectangle labeled "StmtHeader" connected by a line to a dashed rectangular box. Inside the dashed box are several other rounded rectangles, each representing an attribute with its name in bold and a small plus sign (+) in the top right corner.</p> <ul style="list-style-type: none"> proc:stmtDuration proc:stmtEndLabel Label applied to end of statement (referenced by TimeType elements). proc:stmtIsGeneric If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If false then the statement need only be consistent with stmtSpacecraftName (multi-spacecraft missions only). proc:stmtIsMajorEvent Mission specific attribute proc:stmtLabel Label applied to start of statement (referenced by TimeType elements). proc:stmtScheduledTime + Time at which statement should start (if not otherwise constrained). proc:stmtEarliestTime + Planning attribute: Earliest time that statement can be planned to start. proc:stmtLatestTime + Planning attribute: Latest time that statement can be planned to start. proc:stmtSpacecraftName Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only). proc:stmtTitle
namespace	http://www.omg.org/space/procspec
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle
used by	elements BooleanStmts/CheckVariableStmt AllStmts/CmdStmt AllStmts/CommentStmt AllStmts/ControlStmt AllStmts/DirectiveStmt StepType/FORstep/InitFORStmt BooleanStmts/PacketStmt AllStmts/ProcCallStmt AllStmts/SendEventStmt AllStmts/SeqCallStmt AllStmts/SetTelemetryStmt AllStmts/SetVariableStmt StepType/FORstep/TestFORStmt BooleanStmts/TImStmt AllStmts/WaitEventStmt

annotation	documentation Statement header.
source	<pre> <xs:group name="StmtHeader"> <xs:annotation> <xs:documentation>Statement header.</xs:documentation> </xs:annotation> <xs:sequence> <xs:element name="stmtDuration" type="xs:duration" minOccurs="0"/> <xs:element name="stmtEndLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to end of statement (referenced by TimeType elements).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stmtIsGeneric" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If false then the statement need only be consistent with stmtSpacecraftName (multi-spacecraft missions only).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stmtIsMajorEvent" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>Mission specific attribute</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stmtLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to start of statement (referenced by TimeType elements).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="StmtScheduledTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Time at which statement should start (if not otherwise constrained).</xs:documentation> </xs:annotation> </xs:element> <xs:element name="StmtEarliestTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: Earliest time that statement can be planned to start.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="StmtLatestTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: Latest time that statement can be planned to start.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="stmtSpacecraftName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only).</xs:documentation> </xs:annotation> </xs:element></pre>

	<pre><xs:element name="stmtTitle" type="xs:string" minOccurs="0"/> </xs:sequence> </xs:group></pre>
--	---

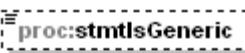
element StmtHeader/stmtDuration

diagram									
namespace	http://www.omg.org/space/procspec								
type	xs:duration								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
source	<pre><xs:element name="stmtDuration" type="xs:duration" minOccurs="0"/></pre>								

element StmtHeader/stmtEndLabel

diagram	 Label applied to end of statement (referenced by TimeType elements).								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	documentation Label applied to end of statement (referenced by TimeType elements).								
source	<pre><xs:element name="stmtEndLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to end of statement (referenced by TimeType elements).</xs:documentation> </xs:annotation> </xs:element></pre>								

element StmtHeader/stmtIsGeneric

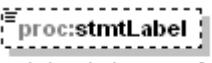
diagram	 If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If false then the statement need only be consistent with stmtSpacecraftName (multi-spacecraft missions only).
namespace	http://www.omg.org/space/procspec

type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If false then the statement need only be consistent with stmtSpacecraftName (multi-spacecraft missions only).
source	<pre><xs:element name="stmtIsGeneric" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If false then the statement need only be consistent with stmtSpacecraftName (multi-spacecraft missions only).</xs:documentation> </xs:annotation> </xs:element></pre>

element StmtHeader/stmtIsMajorEvent

diagram	 Mission specific attribute
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Mission specific attribute
source	<pre><xs:element name="stmtIsMajorEvent" type="xs:boolean" minOccurs="0"> <xs:annotation> <xs:documentation>Mission specific attribute</xs:documentation> </xs:annotation> </xs:element></pre>

element StmtHeader/stmtLabel

diagram	 Label applied to start of statement (referenced by TimeType elements).
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Label applied to start of statement (referenced by TimeType elements).
source	<pre><xs:element name="stmtLabel" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Label applied to start of statement (referenced by TimeType</pre>

	elements).</xs:documentation> </xs:annotation> </xs:element>
--	--

element StmtHeader/StmtScheduledTime

diagram	<pre> classDiagram class proc:TimeType { proc:StmtScheduledTime proc:absoluteTime proc:negativeSign proc:relativeTime } class proc:StmtScheduledTime { <<Time at which statement should start (if not otherwise constrained).>> } class proc:absoluteTime { <<An absolute time literal>> } class proc:absoluteTimeLabel { <<A label defined in a Step or Statement>> } class proc:absoluteTimeVariable { <<A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.>> } class proc:negativeSign { <<proc:negativeSign>> } class proc:relativeTime { <<A relative time literal>> } class proc:relativeTimeVariable { <<A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.>> } proc:StmtScheduledTime --> proc:TimeType proc:TimeType --> proc:absoluteTime proc:TimeType --> proc:negativeSign proc:TimeType --> proc:relativeTime </pre>								
namespace	http://www.omg.org/space/procspec								
type	proc:TimeType								
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable								
annotation	documentation Time at which statement should start (if not otherwise constrained).								
source	<pre> <xs:element name="StmtScheduledTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Time at which statement should start (if not otherwise constrained).</xs:documentation> </xs:annotation> </xs:element> </pre>								

element StmtHeader/StmtEarliestTime

diagram	<pre> classDiagram class proc:TimeType { <<proc:stmtEarliestTime>> <<proc:absoluteTime>> <<proc:label>> <<proc:variable>> <<proc:negativeSign>> <<proc:relativeTime>> } proc:stmtEarliestTime --> proc:TimeType proc:absoluteTime --> proc:TimeType proc:label --> proc:TimeType proc:variable --> proc:TimeType proc:negativeSign --> proc:TimeType proc:relativeTime --> proc:TimeType </pre> <p>The diagram illustrates the structure of the <code>proc:TimeType</code> element. It contains several components:</p> <ul style="list-style-type: none"> <code>proc:stmtEarliestTime</code>: A planning attribute representing the earliest time a statement can be planned to start. <code>proc:absoluteTime</code>: An absolute time literal. <code>proc:label</code>: A label defined in a Step or Statement. <code>proc:variable</code>: A variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. <code>proc:negativeSign</code>: A negative sign indicator. <code>proc:relativeTime</code>: A relative time literal. <code>proc:variable</code>: A variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. 								
namespace	http://www.omg.org/space/procspec								
type	proc:TimeType								
properties	<table border="0"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable								
annotation	<p>documentation</p> <p>Planning attribute: Earliest time that statement can be planned to start.</p>								
source	<pre> <xsd:element name="StmtEarliestTime" type="proc:TimeType" minOccurs="0"> <xsd:annotation> <xsd:documentation>Planning attribute: Earliest time that statement can be planned to start.</xsd:documentation> </xsd:annotation> </xsd:element> </pre>								

element StmtHeader/StmtLatestTime

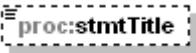
	<pre> graph LR A[proc:stmtLatestTime] --> B(()) B --> C[proc:negativeSign] B --> D[proc:TimeType] D --> E[proc:absoluteTime] D --> F[proc:absoluteTimeLabel] D --> G[proc:absoluteTimeVariable] D --> H[proc:relativeTime] H --> I[proc:relativeTimeVariable] </pre> <p>proc:stmtLatestTime Planning attribute: Latest time that statement can be planned to start.</p> <p>proc:TimeType</p> <ul style="list-style-type: none"> proc:absoluteTime An absolute time literal proc:absoluteTimeLabel A label defined in a Step or Statement proc:absoluteTimeVariable A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. proc:negativeSign proc:relativeTime A relative time literal proc:relativeTimeVariable A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. 								
namespace	http://www.omg.org/space/procspec								
type	proc:TimeType								
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable								
annotation	<p>documentation</p> <p>Planning attribute: Latest time that statement can be planned to start.</p>								
source	<pre> <xs:element name="StmtLatestTime" type="proc:TimeType" minOccurs="0"> <xs:annotation> <xs:documentation>Planning attribute: Latest time that statement can be planned to start.</xs:documentation> </xs:annotation> </xs:element> </pre>								

element StmtHeader/stmtSpacecraftName

	<p>proc:stmtSpacecraftName</p> <p>Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only).</p>
namespace	http://www.omg.org/space/procspec

type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only).
source	<pre><xs:element name="stmtSpacecraftName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only).</xs:documentation> </xs:annotation> </xs:element></pre>

element StmtHeader/stmtTitle

diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
source	<pre><xs:element name="stmtTitle" type="xs:string" minOccurs="0"/></pre>

complexType ExecutionTimeType

diagram	<pre> graph TD ET[ExecutionTimeType] --- > procTimeType[proc:TimeType (extension)] procTimeType --- > procAbsoluteTime[proc:absoluteTime] procTimeType --- > procAbsoluteTimeLabel[proc:absoluteTimeLabel] procTimeType --- > procAbsoluteTimeVariable[proc:absoluteTimeVariable] procTimeType --- > procNegativeSign[proc:negativeSign] procTimeType --- > procRelativeTime[proc:relativeTime] procTimeType --- > procRelativeTimeVariable[proc:relativeTimeVariable] procTimeType --- > procOrbitAngleVariable[proc:orbitAngleVariable] procOrbitAngleVariable --- > procOrbitAngleRadix[proc:orbitAngleRadix] procOrbitAngleVariable --- > procNegativeSign[proc:negativeSign] procOrbitAngleVariable --- > procOrbitAngle[proc:orbitAngle] </pre> <p>The diagram illustrates the structure of the ExecutionTimeType complex type extension. It starts with an ExecutionTimeType element, which is an extension of proc:TimeType. This extension includes several components: proc:absoluteTime, proc:absoluteTimeLabel, proc:absoluteTimeVariable, proc:negativeSign, proc:relativeTime, proc:relativeTimeVariable, and proc:orbitAngleVariable. The proc:orbitAngleVariable component further includes proc:orbitAngleRadix, proc:negativeSign, and proc:orbitAngle.</p>
namespace	http://www.omg.org/space/procspec
type	extension of proc:TimeType
properties	base proc:TimeType
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable proc:orbitAngleVariable proc:orbitAngleRadix proc:negativeSign proc:orbitAngle
used by	element AllStmts/CmdStmt/CmdExecutionTime
annotation	documentation Orbit angle radix followed by value.
source	<x:complexType name="ExecutionTimeType"> <x:annotation>

	<pre> <xs:documentation>Orbit angle radix followed by value.</xs:documentation> </xs:annotation> <xs:complexContent> <xs:extension base="proc:TimeType"> <xs:choice minOccurs="0"> <xs:element name="orbitAngleVariable" type="xs:string"> <xs:annotation> <xs:documentation>A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> <xs:sequence> <xs:element name="orbitAngleRadix"> <xs:annotation> <xs:documentation>A radix for an orbit angle</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="DEG"/> <xs:enumeration value="RAD"/> <xs:enumeration value="OAG"/> </xs:restriction> </xs:simpleType> </xs:sequence> <xs:element name="negativeSign" type="xs:boolean" default="false" minOccurs="0"/> <xs:element name="orbitAngle"> <xs:annotation> <xs:documentation>An orbit angle value. Orbit number is allowed e.g. 1-20 for orbit 1 angle 20.</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="(d+ -)?\d*.?\d*"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:choice> </xs:extension> </xs:complexContent> </xs:complexType> </pre>
--	---

element ExecutionTimeType/orbitAngleVariable

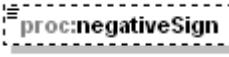
diagram	 <p>A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</p>				
namespace	http://www.omg.org/space/procspes				
type	xs:string				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple
isRef	0				
content	simple				

annotation	<p>documentation A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</p>
source	<pre><xs:element name="orbitAngleVariable" type="xs:string"> <xs:annotation> <xs:documentation>A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element></pre>

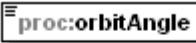
element ExecutionTimeType/orbitAngleRadix

diagram	 <p>A radix for an orbit angle</p>						
namespace	http://www.omg.org/space/procspec						
type	restriction of xs:string						
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple		
isRef	0						
content	simple						
facets	<table> <tr> <td>enumeration</td> <td>DEG</td> </tr> <tr> <td>enumeration</td> <td>RAD</td> </tr> <tr> <td>enumeration</td> <td>OAG</td> </tr> </table>	enumeration	DEG	enumeration	RAD	enumeration	OAG
enumeration	DEG						
enumeration	RAD						
enumeration	OAG						
annotation	<p>documentation A radix for an orbit angle</p>						
source	<pre><xs:element name="orbitAngleRadix"> <xs:annotation> <xs:documentation>A radix for an orbit angle</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="DEG"/> <xs:enumeration value="RAD"/> <xs:enumeration value="OAG"/> </xs:restriction> </xs:simpleType> </xs:element></pre>						

element ExecutionTimeType/negativeSign

diagram											
namespace	http://www.omg.org/space/procspec										
type	xs:boolean										
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> <tr> <td>default</td> <td>false</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	false
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	false										
source	<pre><xs:element name="negativeSign" type="xs:boolean" default="false" minOccurs="0"/></pre>										

element **ExecutionTimeType/orbitAngle**

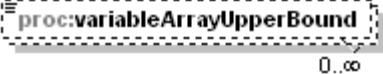
diagram	 proc:orbitAngle An orbit angle value. Orbit number is allowed e.g. 1-20 for orbit 1 angle 20.
namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
properties	isRef 0 content simple
facets	pattern <code>(\d+ -)?\d*\.\?\d*</code>
annotation	documentation An orbit angle value. Orbit number is allowed e.g. 1-20 for orbit 1 angle 20.
source	<pre><xs:element name="orbitAngle"> <xs:annotation> <xs:documentation>An orbit angle value. Orbit number is allowed e.g. 1-20 for orbit 1 angle 20.</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="(\d+ -)?\d*\.\?\d*"/> </xs:restriction> </xs:simpleType> </xs:element></pre>

complexType **ProcedureVariable**

diagram	
namespace	http://www.omg.org/space/procspec

type	extension of proc:Variable
properties	base proc:Variable
children	proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut proc:variableArrayUpperBound
used by	element Proc/Variable
source	<pre><xs:complexType name="ProcedureVariable"> <xs:complexContent> <xs:extension base="proc:Variable"> <xs:sequence> <xs:element name="variableArrayUpperBound" type="xs:int" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>If specified declares variable as an Array by specifying each of its dimensions. Not supported for DERIVED (VariableTypeEnum) types.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType></pre>

element ProcedureVariable/variableArrayUpperBound

diagram	 <p>If specified declares variable as an Array by specifying each of its dimensions. Not supported for DERIVED (VariableTypeEnum) types.</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:int								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	unbounded	content	simple
isRef	0								
minOcc	0								
maxOcc	unbounded								
content	simple								
annotation	<p>documentation</p> <p>If specified declares variable as an Array by specifying each of its dimensions. Not supported for DERIVED (VariableTypeEnum) types.</p>								
source	<pre><xs:element name="variableArrayUpperBound" type="xs:int" minOccurs="0" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>If specified declares variable as an Array by specifying each of its dimensions. Not supported for DERIVED (VariableTypeEnum) types.</xs:documentation> </xs:annotation> </xs:element></pre>								

complexType **SequenceVariable**

diagram	<pre> classDiagram class SequenceVariable { <<proc:Variable>> proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut proc:rangeSetOverride } SequenceVariable < -- proc:Variable </pre>
namespace	http://www.omg.org/space/procspec
type	extension of proc:Variable

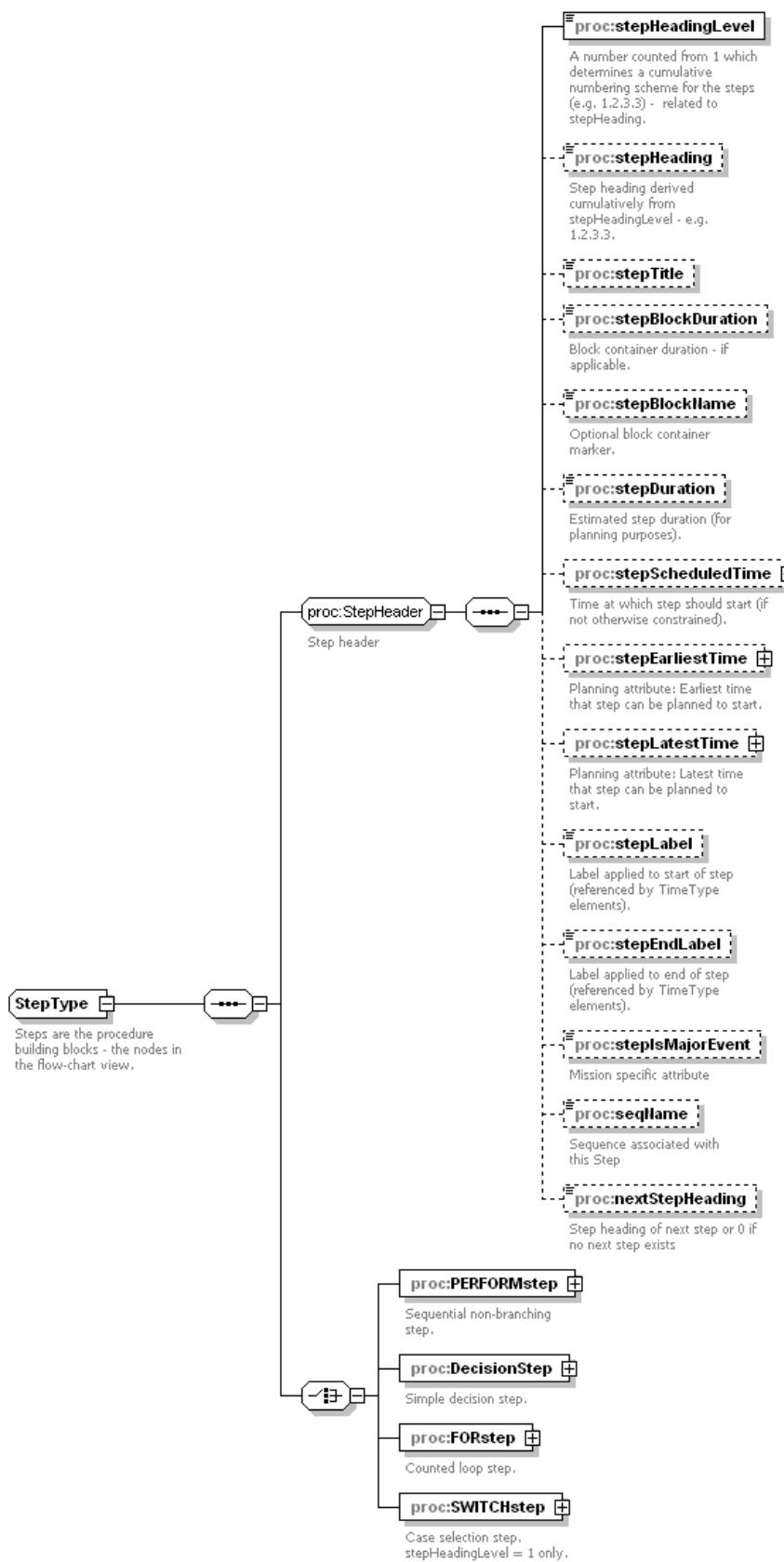
properties	base proc:Variable
children	proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut proc:rangeSetOverride
used by	element Proc/Sequence/SeqVariable
source	<pre><xs:complexType name="SequenceVariable"> <xs:complexContent> <xs:extension base="proc:Variable"> <xs:sequence> <xs:element name="rangeSetOverride" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Range set override (must be compatible with referenced CmdParam or SeqParam)</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:extension> </xs:complexContent> </xs:complexType></pre>

element SequenceVariable/rangeSetOverride

diagram	 <p>Range set override (must be compatible with referenced CmdParam or SeqParam)</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>Range set override (must be compatible with referenced CmdParam or SeqParam)</p>								
source	<pre><xs:element name="rangeSetOverride" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Range set override (must be compatible with referenced CmdParam or SeqParam)</xs:documentation> </xs:annotation> </xs:element></pre>								

complexType **StepType**

diagram



namespace	http://www.omg.org/space/procspec
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep
used by	elements Proc/ProcBody/Step StepType/DecisionStep/IF/THEN/Step StepType/DecisionStep/IF/ELSE/Step StepType/DecisionStep/WHILE/REPEAT/Step StepType/DecisionStep/REPEAT_UNTIL/REPEAT/Step StepType/DecisionStep/WAIT/CONTINUE/Step StepType/DecisionStep/WAIT/TIMEOUT/Step StepType/DecisionStep/PRECONDITION/THEN/Step StepType/FORstep/FOR/REPEAT/Step StepType/SWITCHstep/Case/CaseExecute/Step
annotation	documentation Steps are the procedure building blocks - the nodes in the flow-chart view.
source	<pre> <xs:complexType name="StepType"> <xs:annotation> <xs:documentation>Steps are the procedure building blocks - the nodes in the flow-chart view.</xs:documentation> </xs:annotation> <xs:sequence> <xs:group ref="proc:StepHeader"> <xs:annotation> <xs:documentation>Step header</xs:documentation> </xs:annotation> </xs:group> <xs:choice> <xs:element name="PERFORMstep"> <xs:annotation> <xs:documentation>Sequential non-branching step.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:group ref="proc:AllStmts"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="DecisionStep"> <xs:annotation> <xs:documentation>Simple decision step.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:choice> <xs:element name="ArithmeticResult"> <xs:annotation> <xs:documentation>The (boolean) result of an arithmetic expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element> <xs:element name="BooleanResult"> <xs:annotation> <xs:documentation>The result of booleanExpression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:BooleanResult"/> </xs:complexType> </xs:element> </xs:choice> </xs:sequence> </xs:complexType> </xs:element> </xs:choice> </xs:sequence> </xs:complexType> </pre>

```

</xs:complexType>
</xs:element>
</xs:choice>
<xs:choice>
<xs:annotation>
<xs:documentation>Steps branching on booleanResult</xs:documentation>
</xs:annotation>
<xs:element name="IF">
<xs:annotation>
<xs:documentation>IF-THEN-ELSE branch. stepHeadingLevel = 1
only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="THEN">
<xs:annotation>
<xs:documentation>Branch taken when booleanResult is True.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="THENBranchName" minOccurs="0">
<xs:annotation>
<xs:documentation>For display purposes only. Valid pairs are then/else, yes/no,
y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ELSE" minOccurs="0">
<xs:annotation>
<xs:documentation>Branch taken when booleanResult is
False.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="ELSEBranchName" minOccurs="0">
<xs:annotation>
<xs:documentation>For display purposes only. Valid pairs are then/else, yes/no,
y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="WHILE">
<xs:annotation>
<xs:documentation>Repeats as long as booleanResult is True. stepHeadingLevel = 1
only.</xs:documentation>
</xs:annotation>
<xs:complexType>

```

```

<xs:sequence>
<xs:element name="REPEAT">
<xs:complexType>
<xs:sequence>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="REPEAT_UNTIL">
<xs:annotation>
<xs:documentation>Repeats until booleanResult is True. The test is made at the end of
the loop (so it executes at least once). stepHeadingLevel = 1 only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="REPEAT">
<xs:complexType>
<xs:sequence>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="WAIT">
<xs:annotation>
<xs:documentation>Waits until booleanResult is True.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="waitPollingPeriod" type="xs:duration">
<xs:annotation>
<xs:documentation>Period between booleanResult tests</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="waitForTimeoutPeriod" type="xs:duration" minOccurs="0">
<xs:annotation>
<xs:documentation>Timeout period</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="CONTINUE" minOccurs="0">
<xs:annotation>
<xs:documentation>Executed when booleanResult is True. Only useful if there are
TIMEOUT steps. stepHeadingLevel = 1 only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TIMEOUT" minOccurs="0">

```

```

<xs:annotation>
  <xs:documentation>Executed on Timeout instead of CONTINUE steps.
stepHeadingLevel = 1 only. </xs:documentation>
</xs:annotation>
<xs:complexType>
  <xs:sequence>
    <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="continueOnTimeout" type="xs:boolean" default="false"
minOccurs="0">
  <xs:annotation>
    <xs:documentation>If false and no TIMEOUT steps then terminate execution on
timeout. stepHeadingLevel = 1 only.</xs:documentation>
  </xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="PRECONDITION">
  <xs:annotation>
    <xs:documentation>IF-THEN branch. stepHeadingLevel > 1
only.</xs:documentation>
  </xs:annotation>
<xs:complexType>
  <xs:sequence>
    <xs:element name="THEN">
      <xs:annotation>
        <xs:documentation>Branch taken when booleanResult is True.</xs:documentation>
      </xs:annotation>
<xs:complexType>
  <xs:sequence>
    <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="FORstep">
  <xs:annotation>
    <xs:documentation>Counted loop step.</xs:documentation>
  </xs:annotation>
<xs:complexType>
  <xs:sequence>
    <xs:element name="LoopVariableName" type="xs:string">
      <xs:annotation>
        <xs:documentation>Name of integer loop variable to be checked. If a Variable array then
the array index is enclosed by one or more brackets () directly after the
variable.</xs:documentation>
      </xs:annotation>

```

```

</xs:element>
<xs:element name="LoopVariableIncrement" type="xs:integer">
<xs:annotation>
<xs:documentation>Value by which the loop variable is incremented (non-zero)</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="InitFORStmt">
<xs:annotation>
<xs:documentation>Statement to initialise the loop Variable</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:group ref="proc:StmtHeader"/>
<xs:group ref="proc:SetVarValue"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TestFORStmt">
<xs:annotation>
<xs:documentation>Statement to check the loop variable. If true the variable is incremented by LoopVariableIncrement and the loop repeated</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:group ref="proc:StmtHeader"/>
<xs:group ref="proc:CheckVarValue"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="FOR">
<xs:annotation>
<xs:documentation>Repeats as long as TestFORStmt is True. stepHeadingLevel = 1 only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="REPEAT">
<xs:complexType>
<xs:sequence>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="SWITCHstep">
<xs:annotation>
<xs:documentation>Case selection step. stepHeadingLevel = 1 only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>

```

```

<xs:element name="Case" maxOccurs="unbounded">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="caseBranchName" type="xs:string" minOccurs="0">
        <xs:annotation>
          <xs:documentation>Name of the branch - for display purposes
only.</xs:documentation>
        </xs:annotation>
      </xs:element>
      <xs:choice minOccurs="0">
        <xs:element name="CaseArithmeticResult">
          <xs:annotation>
            <xs:documentation>The (boolean) result of an arithmetic
expression</xs:documentation>
          </xs:annotation>
          <xs:complexType>
            <xs:group ref="proc:ArithmeticResult"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="CaseBooleanResult">
          <xs:annotation>
            <xs:documentation>The result of booleanExpression</xs:documentation>
          </xs:annotation>
          <xs:complexType>
            <xs:group ref="proc:BooleanResult"/>
          </xs:complexType>
        </xs:element>
      </xs:choice>
      <xs:element name="CaseExecute">
        <xs:annotation>
          <xs:documentation>Executed if CaseBooleanResult is true.</xs:documentation>
        </xs:annotation>
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:sequence>
        <xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:choice>
</xs:sequence>
</xs:complexType>

```

element StepType/PERFORMstep

diagram	<pre> classDiagram class proc::PERFORMstep class proc::AllStmts class proc::CmdStmt class proc::ProcCallStmt class proc::CommentStmt class proc::ControlStmt class proc::DirectiveStmt class proc::SetVariableStmt class proc::SetTelemetryStmt class proc::SendEventStmt class proc::WaitEventStmt class proc::SeqCallStmt class proc::BooleanStmts class proc::TlmStmt class proc::CheckVariableStmt class proc::PacketStmt proc::PERFORMstep --o proc::AllStmts proc::AllStmts --o+ +proc::CmdStmt +proc::ProcCallStmt +proc::CommentStmt +proc::ControlStmt +proc::DirectiveStmt +proc::SetVariableStmt +proc::SetTelemetryStmt +proc::SendEventStmt +proc::WaitEventStmt +proc::SeqCallStmt +proc::BooleanStmts +proc::TlmStmt +proc::CheckVariableStmt +proc::PacketStmt </pre> <p>Sequential non-branching step.</p> <p>0..∞ All Statements.</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table border="0"> <tr> <td>isRef</td><td>0</td> </tr> <tr> <td>content</td><td>complex</td> </tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:CmdStmt proc:ProcCallStmt proc:CommentStmt proc:ControlStmt proc:DirectiveStmt proc:SetVariableStmt proc:SetTelemetryStmt proc:SendEventStmt proc:WaitEventStmt proc:SeqCallStmt proc:TlmStmt proc:CheckVariableStmt proc:PacketStmt				

annotation	documentation Sequential non-branching step.
source	<pre><xs:element name="PERFORMstep"> <xs:annotation> <xs:documentation>Sequential non-branching step.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence minOccurs="0" maxOccurs="unbounded"> <xs:group ref="proc:AllStmts"/> </xs:sequence> </xs:complexType> </xs:element></pre>

element **StepType/DecisionStep**

diagram	<p>Simple decision step.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:ArithmeticResult proc:BooleanResult proc:IF proc:WHILE proc:REPEAT_UNTIL proc:WAIT proc:PRECONDITION
annotation	documentation Simple decision step.
source	<pre><xs:element name="DecisionStep"> <xs:annotation></pre>

	<pre> <xs:documentation>Simple decision step.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:choice> <xs:element name="ArithmeticResult"> <xs:annotation> <xs:documentation>The (boolean) result of an arithmetic expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element> <xs:element name="BooleanResult"> <xs:annotation> <xs:documentation>The result of booleanExpression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:BooleanResult"/> </xs:complexType> </xs:element> </xs:choice> <xs:choice> <xs:annotation> <xs:documentation>Steps branching on booleanResult</xs:documentation> </xs:annotation> <xs:element name="IF"> <xs:annotation> <xs:documentation>IF-THEN-ELSE branch. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="THEN"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is True.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="THENBranchName" minOccurs="0"> <xs:annotation> <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="ELSE" minOccurs="0"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is False.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="ELSEBranchName" minOccurs="0"> </pre>
--	---

```

<xs:annotation>
  <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no,
y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="WHILE">
<xs:annotation>
  <xs:documentation>Repeats as long as booleanResult is True. stepHeadingLevel = 1
only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
  <xs:element name="REPEAT">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="REPEAT_UNTIL">
<xs:annotation>
  <xs:documentation>Repeats until booleanResult is True. The test is made at the end of the
loop (so it executes at least once). stepHeadingLevel = 1 only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
  <xs:element name="REPEAT">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="WAIT">
<xs:annotation>
  <xs:documentation>Waits until booleanResult is True.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
  <xs:element name="waitPollingPeriod" type="xs:duration">
    <xs:annotation>
      <xs:documentation>Period between booleanResult tests</xs:documentation>
    </xs:annotation>
  </xs:element>
</xs:sequence>
</xs:complexType>

```

```

</xs:element>
<xs:element name="waitForTimeoutPeriod" type="xs:duration" minOccurs="0">
<xs:annotation>
<xs:documentation>Timeout period</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="CONTINUE" minOccurs="0">
<xs:annotation>
<xs:documentation>Executed when booleanResult is True. Only useful if there are
TIMEOUT steps. stepHeadingLevel = 1 only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="TIMEOUT" minOccurs="0">
<xs:annotation>
<xs:documentation>Executed on Timeout instead of CONTINUE steps.
stepHeadingLevel = 1 only. </xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="continueOnTimeout" type="xs:boolean" default="false"
minOccurs="0">
<xs:annotation>
<xs:documentation>If false and no TIMEOUT steps then terminate execution on
timeout. stepHeadingLevel = 1 only.</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="PRECONDITION">
<xs:annotation>
<xs:documentation>IF-THEN branch. stepHeadingLevel > 1 only.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="THEN">
<xs:annotation>
<xs:documentation>Branch taken when booleanResult is True.</xs:documentation>
</xs:annotation>
<xs:complexType>
<xs:sequence>
<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>

```

	<pre> </xs:element> </xs:choice> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

element StepType/DecisionStep/ArithmeticResult

diagram	<p>The (boolean) result of an arithmetic expression</p> <p>The arithmetic result of an expression.</p> <p>proc:UnaryOperator Applies a unary operator minus (-) or NOT to the bracketed expression</p> <p>proc:Bracket Brackets the following expression</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:UnaryOperator proc:Bracket
annotation	documentation The (boolean) result of an arithmetic expression
source	<pre> <xs:element name="ArithmeticResult"> <xs:annotation> <xs:documentation>The (boolean) result of an arithmetic expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element> </pre>

element StepType/DecisionStep/BooleanResult

diagram	<p>The result of booleanExpression</p> <p>The Boolean result of an expression.</p> <p>proc:NOT NOTs the bracketed expression</p> <p>proc:Bracket Brackets the following expression</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:NOT proc:Bracket
annotation	documentation The result of booleanExpression
source	<pre> <xs:element name="BooleanResult"> <xs:annotation> <xs:documentation>The result of booleanExpression</xs:documentation> </xs:annotation> </pre>

	<pre><xs:complexType> <xs:group ref="proc:BooleanResult"/> </xs:complexType> </xs:element></pre>
--	--

element StepType/DecisionStep/IF

diagram	<p>IF-THEN-ELSE branch. stepHeadingLevel = 1 only.</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:THEN proc:ELSE				
annotation	<p>documentation IF-THEN-ELSE branch. stepHeadingLevel = 1 only.</p>				
source	<pre><xs:element name="IF"> <xs:annotation> <xs:documentation>IF-THEN-ELSE branch. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="THEN"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is True.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="THENBranchName" minOccurs="0"> <xs:annotation> <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"> <xs:annotation> <xs:documentation>proc:StepType</xs:documentation> </xs:annotation> </xs:element> <xs:element name="ELSE" minOccurs="0"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is False.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="ELSEBranchName" minOccurs="0"> <xs:annotation> <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType></pre>				

	<pre> </xs:element> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	---

element StepType/DecisionStep/IF/THEN

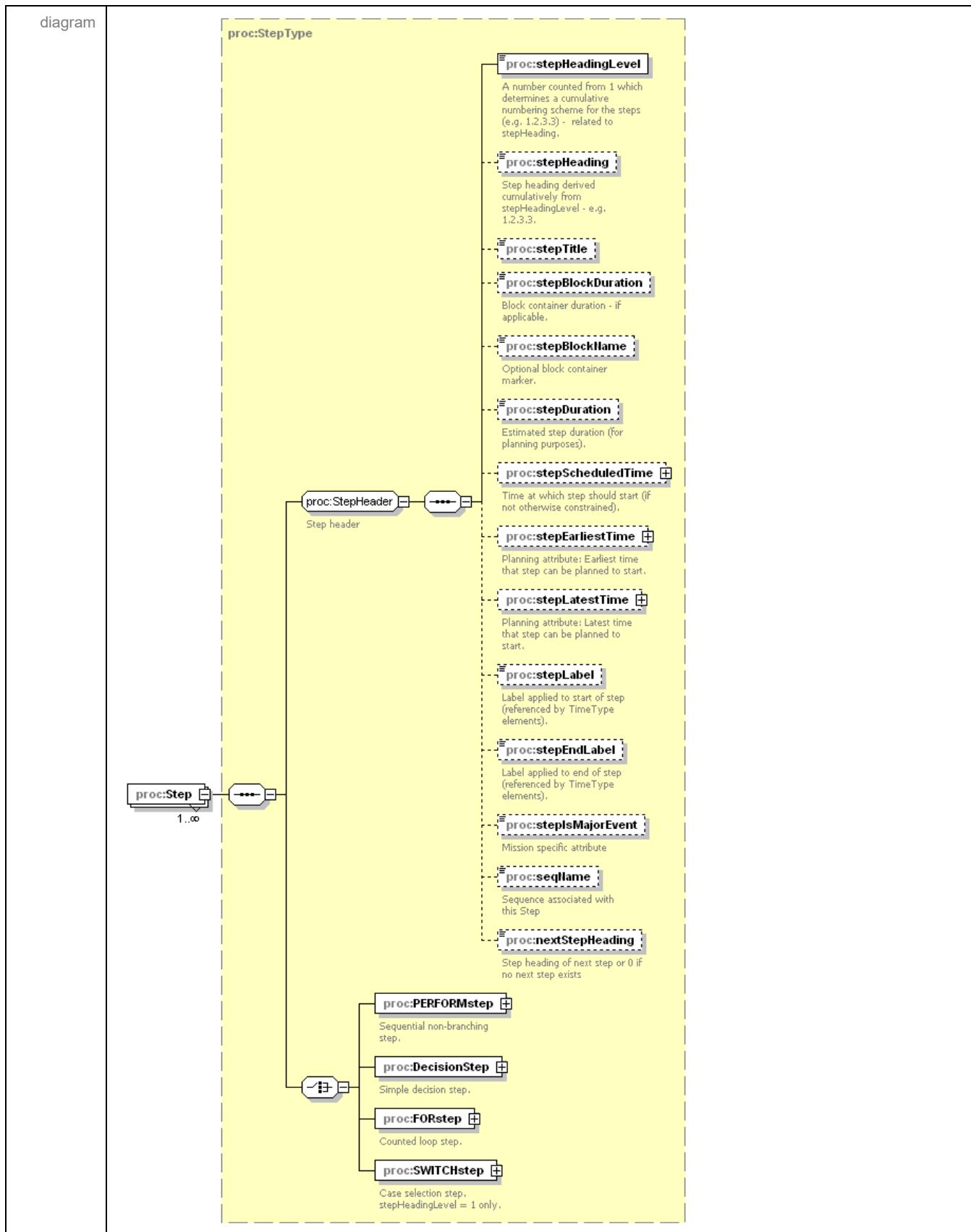
diagram	<p>proc:THEN</p> <p>Branch taken when booleanResult is True.</p> <p>proc:THENBranchName</p> <p>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</p> <p>proc:Step</p> <p>1..∞</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:THENBranchName proc:Step				
annotation	<p>documentation</p> <p>Branch taken when booleanResult is True.</p>				
source	<pre> <xs:element name="THEN"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is True.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="THENBranchName" minOccurs="0"> <xs:annotation> <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </pre>				

element StepType/DecisionStep/IF/THEN/THENBranchName

diagram	<p>proc:THENBranchName</p> <p>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</p>
namespace	http://www.omg.org/space/procspec

properties	isRef 0 minOcc 0 maxOcc 1
annotation	documentation For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.
source	<xs:element name="THENBranchName" minOccurs="0"> <xs:annotation> <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation> </xs:annotation> </xs:element>

element **StepType/DecisionStep/IF/THEN/Step**

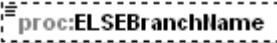


namespace	http://www.omg.org/space/procspec
type	proc:StepType
properties	isRef 0 minOcc 1 maxOcc unbounded content complex
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>

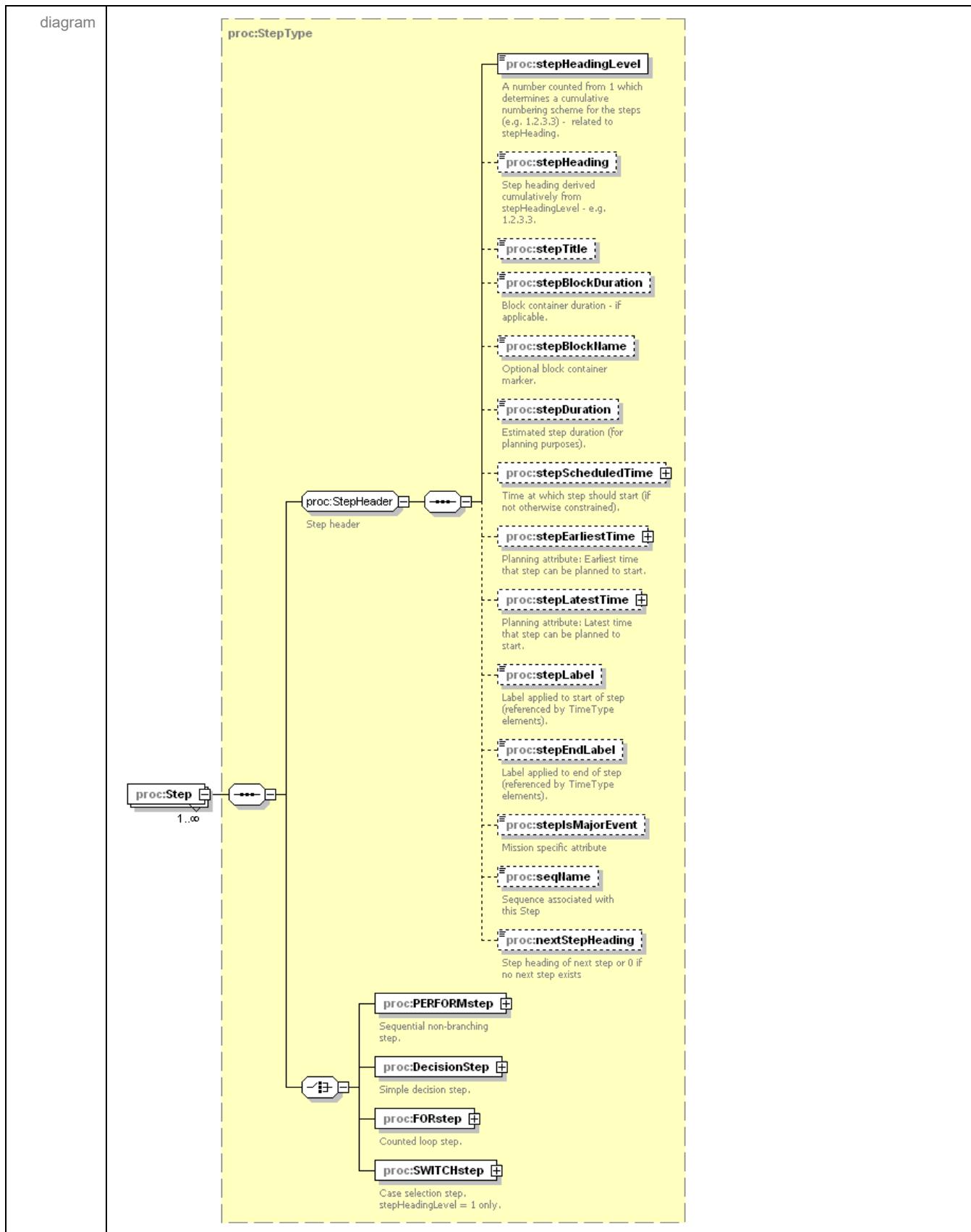
element StepType/DecisionStep/IF/ELSE

diagram	<p>proc:ELSEBranchName</p> <p>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</p> <p>proc:Step 1..oo</p> <p>Branch taken when booleanResult is False.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc 1 content complex
children	proc:ELSEBranchName proc:Step
annotation	documentation Branch taken when booleanResult is False.
source	<xs:element name="ELSE" minOccurs="0"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is False.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="ELSEBranchName" minOccurs="0"> <xs:annotation> <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation> </xs:annotation> </xs:element> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element>

element **StepType/DecisionStep/IF/ELSE/ELSEBranchName**

diagram							
	For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.						
namespace	http://www.omg.org/space/procspec						
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1
isRef	0						
minOcc	0						
maxOcc	1						
annotation	<p>documentation For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</p>						
source	<pre><xs:element name="ELSEBranchName" minOccurs="0"> <xs:annotation> <xs:documentation>For display purposes only. Valid pairs are then/else, yes/no, y/n, true/false, ok/nok, go/nogo and valid/invalid.</xs:documentation> </xs:annotation> </xs:element></pre>						

element **StepType/DecisionStep/IF/ELSE/Step**



namespace	http://www.omg.org/space/procspec								
type	proc:StepType								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>1</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	1	maxOcc	unbounded	content	complex
isRef	0								
minOcc	1								
maxOcc	unbounded								
content	complex								
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep								
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>								

element StepType/DecisionStep/WILE

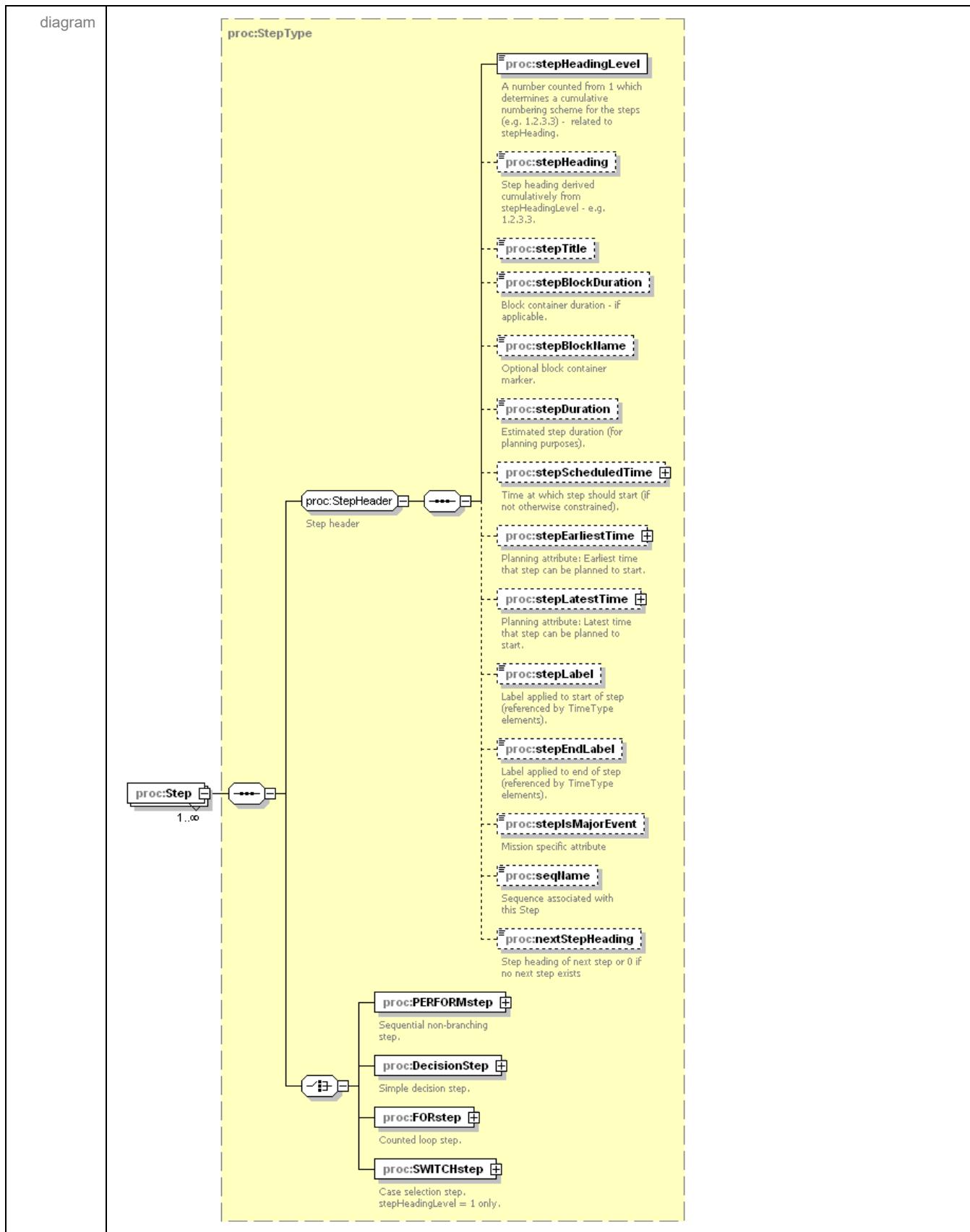
diagram	<p>Repeats as long as booleanResult is True, stepHeadingLevel = 1 only.</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:REPEAT				
annotation	<p>documentation</p> <p>Repeats as long as booleanResult is True. stepHeadingLevel = 1 only.</p>				
source	<xs:element name="WHILE"> <xs:annotation> <xs:documentation>Repeats as long as booleanResult is True. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="REPEAT"> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element>				

element StepType/DecisionStep/WILE/REPEAT

diagram					
namespace	http://www.omg.org/space/procspec				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				

children	<u>proc:Step</u>
source	<xs:element name="REPEAT"> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element>

element **StepType/DecisionStep/WHILE/REPEAT/Step**



namespace	http://www.omg.org/space/procspec								
type	proc:StepType								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>1</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	1	maxOcc	unbounded	content	complex
isRef	0								
minOcc	1								
maxOcc	unbounded								
content	complex								
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep								
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>								

element StepType/DecisionStep/REPEAT_UNTIL

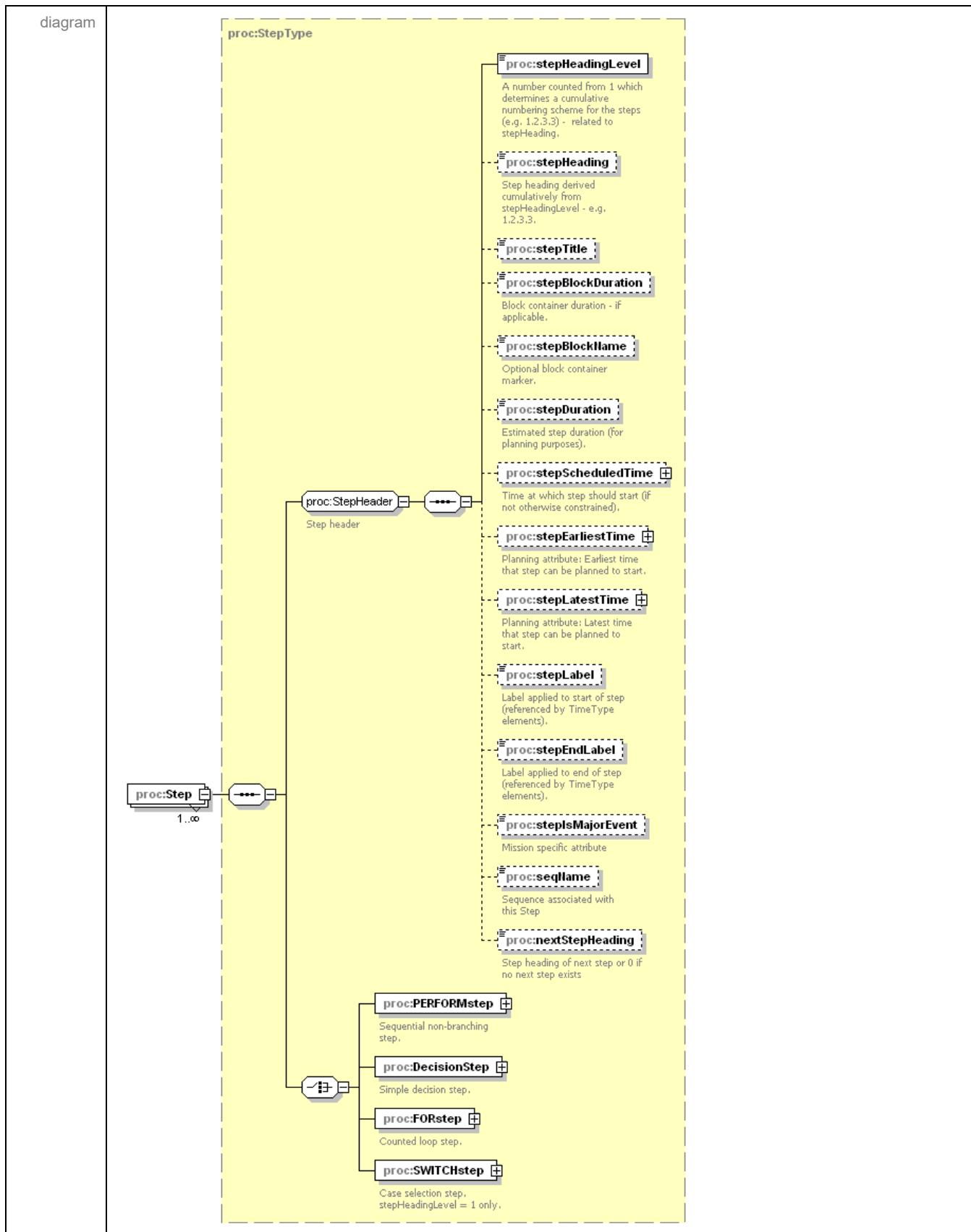
diagram	<p>Repeats until booleanResult is True. The test is made at the end of the loop (so it executes at least once). stepHeadingLevel = 1 only.</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:REPEAT				
annotation	<p>documentation</p> <p>Repeats until booleanResult is True. The test is made at the end of the loop (so it executes at least once). stepHeadingLevel = 1 only.</p>				
source	<xs:element name="REPEAT_UNTIL"> <xs:annotation> <xs:documentation>Repeats until booleanResult is True. The test is made at the end of the loop (so it executes at least once). stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element>				

element StepType/DecisionStep/REPEAT_UNTIL/REPEAT

diagram	<p>1..∞</p>
namespace	http://www.omg.org/space/procspec

properties	isRef 0 content complex
children	<u>proc:Step</u>
source	<xs:element name="REPEAT"> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element>

element **StepType/DecisionStep/REPEAT_UNTIL/REPEAT/Step**



namespace	http://www.omg.org/space/procspec
type	proc:StepType
properties	isRef 0 minOcc 1 maxOcc unbounded content complex
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>

element StepType/DecisionStep/WAIT

diagram	<p>The diagram illustrates the execution flow of a WAIT step. It starts with a proc:WAIT step, followed by a decision point. From this decision point, four paths emerge:</p> <ul style="list-style-type: none"> proc:waitPollingPeriod: Period between booleanResult tests. proc:waitTimeoutPeriod: Timeout period. proc:CONTINUE: Executed when booleanResult is True. Only useful if there are TIMEOUT steps, stepHeadingLevel = 1 only. proc:TIMEOUT: Executed on Timeout instead of CONTINUE steps, stepHeadingLevel = 1 only. <p>If the booleanResult is False and no TIMEOUT steps are present, the execution terminates on timeout, with stepHeadingLevel = 1 only.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:waitPollingPeriod proc:waitTimeoutPeriod proc:CONTINUE proc:TIMEOUT proc:continueOnTimeout
annotation	documentation Waits until booleanResult is True.
source	<pre> <xs:element name="WAIT"> <xs:annotation> <xs:documentation>Waits until booleanResult is True.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="waitPollingPeriod" type="xs:duration"> <xs:annotation> <xs:documentation>Period between booleanResult tests</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>

	<pre> </xs:element> <xs:element name="waitForTimeoutPeriod" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout period</xs:documentation> </xs:annotation> </xs:element> <xs:element name="CONTINUE" minOccurs="0"> <xs:annotation> <xs:documentation>Executed when booleanResult is True. Only useful if there are TIMEOUT steps. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="TIMEOUT" minOccurs="0"> <xs:annotation> <xs:documentation>Executed on Timeout instead of CONTINUE steps. stepHeadingLevel = 1 only. </xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="continueOnTimeout" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If false and no TIMEOUT steps then terminate execution on timeout. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

element StepType/DecisionStep/WAIT/waitPollingPeriod

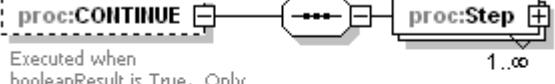
diagram	 <p>Period between booleanResult tests</p>
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 content simple
annotation	<p>documentation</p> <p>Period between booleanResult tests</p>
source	<pre> <xs:element name="waitForTimeoutPeriod" type="xs:duration"> <xs:annotation> <xs:documentation>Period between booleanResult tests</xs:documentation> </xs:annotation> </pre>

	</xs:element>
--	---------------

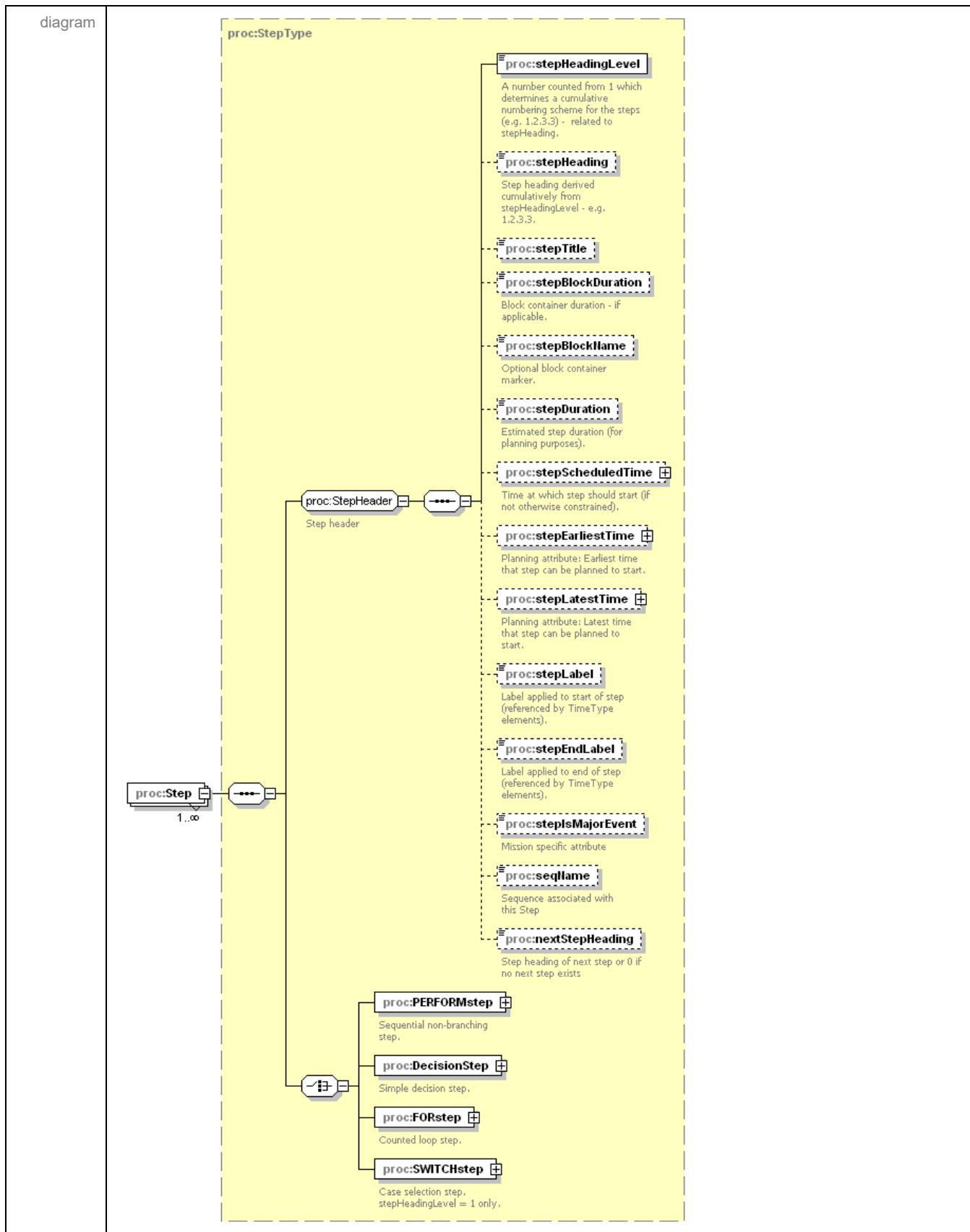
element **StepType/DecisionStep/WAIT/waitTimeoutPeriod**

diagram	 <p>proc:waitTimeoutPeriod Timeout period</p>
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 minOcc 0 maxOcc 1 content simple
annotation	documentation Timeout period
source	<pre><xs:element name="waitTimeoutPeriod" type="xs:duration" minOccurs="0"> <xs:annotation> <xs:documentation>Timeout period</xs:documentation> </xs:annotation> </xs:element></pre>

element **StepType/DecisionStep/WAIT/CONTINUE**

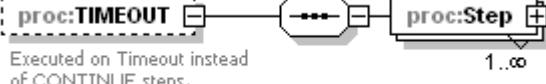
diagram	 <p>proc:CONTINUE Executed when booleanResult is True. Only useful if there are TIMEOUT steps. stepHeadingLevel = 1 only.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 minOcc 0 maxOcc 1 content complex
children	proc:Step
annotation	documentation Executed when booleanResult is True. Only useful if there are TIMEOUT steps. stepHeadingLevel = 1 only.
source	<pre><xs:element name="CONTINUE" minOccurs="0"> <xs:annotation> <xs:documentation>Executed when booleanResult is True. Only useful if there are TIMEOUT steps. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element></pre>

element **StepType/DecisionStep/WAIT/CONTINUE/Step**

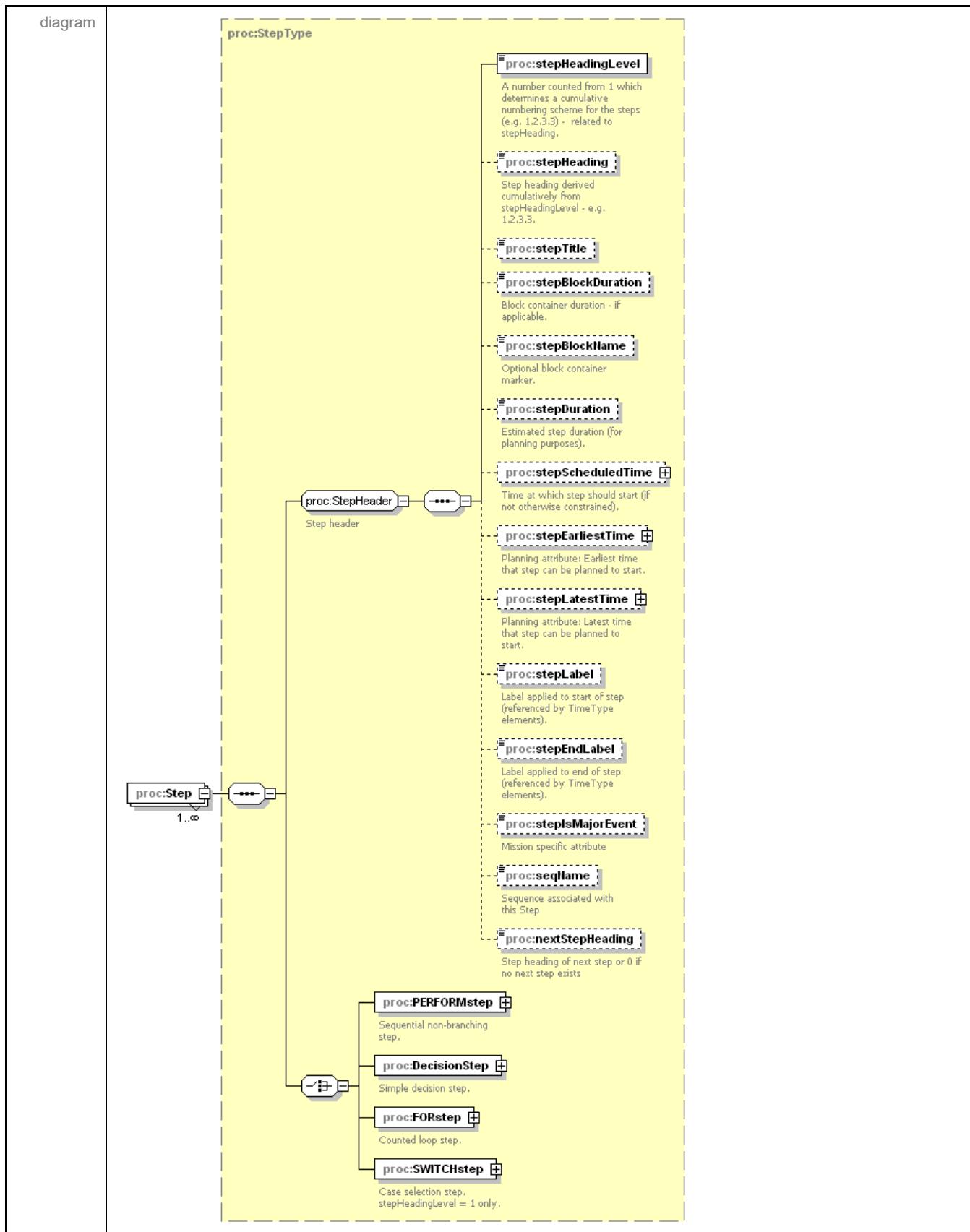


namespace	http://www.omg.org/space/procspec								
type	proc:StepType								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>1</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	1	maxOcc	unbounded	content	complex
isRef	0								
minOcc	1								
maxOcc	unbounded								
content	complex								
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep								
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>								

element StepType/DecisionStep/WAIT/TIMEOUT

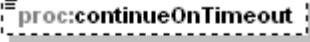
diagram	 <p>Executed on Timeout instead of CONTINUE steps, stepHeadingLevel = 1 only.</p>								
namespace	http://www.omg.org/space/procspec								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>0</td> </tr> <tr> <td>maxOcc</td> <td>1</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	complex
isRef	0								
minOcc	0								
maxOcc	1								
content	complex								
children	proc:Step								
annotation	<p>documentation</p> <p>Executed on Timeout instead of CONTINUE steps. stepHeadingLevel = 1 only.</p>								
source	<xs:element name="TIMEOUT" minOccurs="0"> <xs:annotation> <xs:documentation>Executed on Timeout instead of CONTINUE steps. stepHeadingLevel = 1 only. </xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element>								

element **StepType/DecisionStep/WAIT/TIMEOUT/Step**



namespace	http://www.omg.org/space/procspec
type	proc:StepType
properties	isRef 0 minOcc 1 maxOcc unbounded content complex
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>

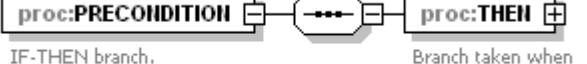
element StepType/DecisionStep/WAIT/continueOnTimeout

diagram	 If false and no TIMEOUT steps then terminate execution on timeout. stepHeadingLevel = 1 only.
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple default false

annotation documentation
If false and no TIMEOUT steps then terminate execution on timeout. stepHeadingLevel = 1 only.

source	<xs:element name="continueOnTimeout" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>If false and no TIMEOUT steps then terminate execution on timeout. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> </xs:element>
--------	--

element StepType/DecisionStep/PRECONDITION

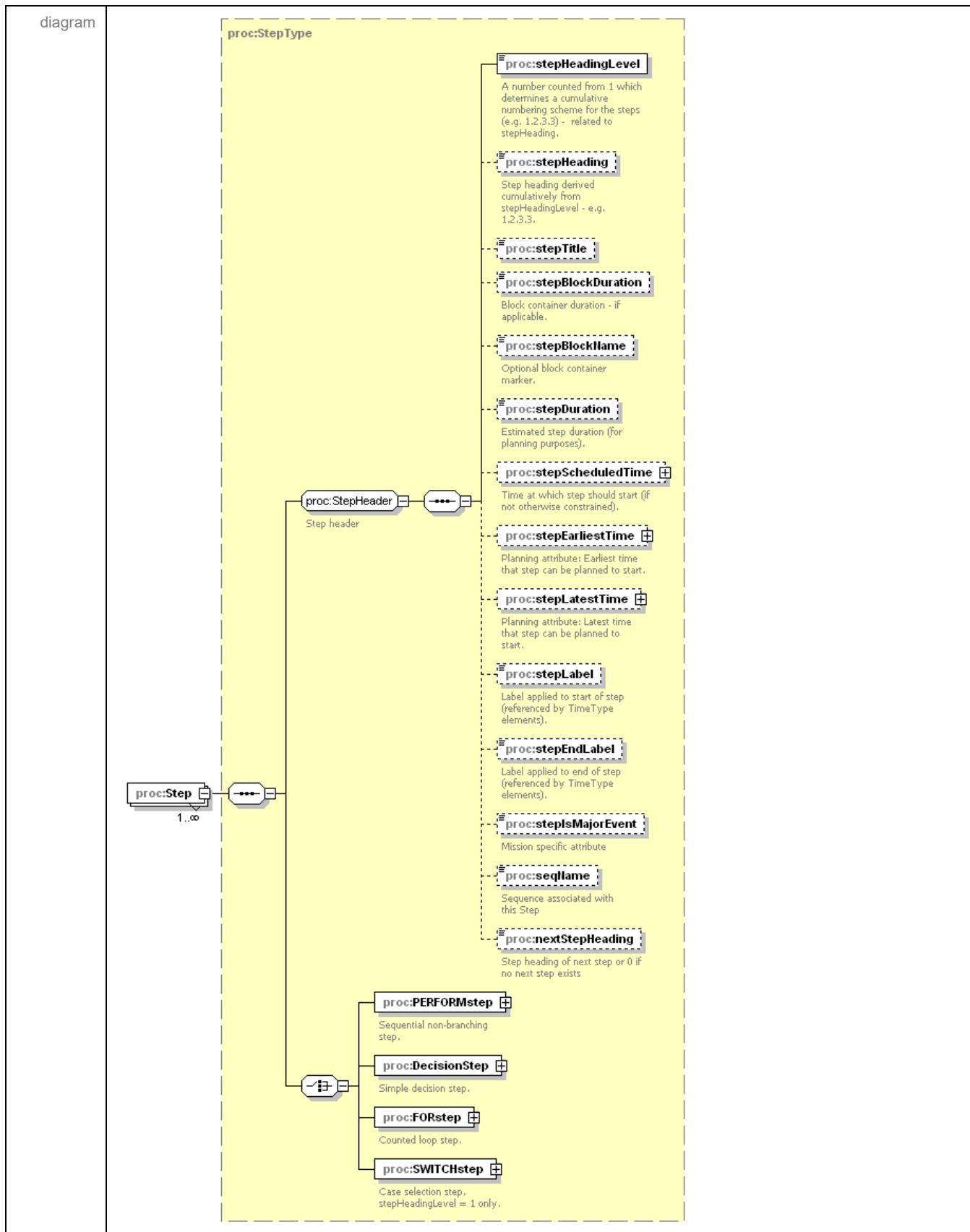
diagram	 IF-THEN branch. stepHeadingLevel > 1 only. Branch taken when booleanResult is True.
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:THEN
annotation	documentation IF-THEN branch. stepHeadingLevel > 1 only.
source	<xs:element name="PRECONDITION"> <xs:annotation> <xs:documentation>IF-THEN branch. stepHeadingLevel > 1 only.</xs:documentation> </xs:annotation> <xs:complexType>

	<pre> <xs:sequence> <xs:element name="THEN"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is True.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

element StepType/DecisionStep/PRECONDITION/THEN

diagram	<p>Branch taken when booleanResult is True.</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:Step				
annotation	<p>documentation Branch taken when booleanResult is True.</p>				
source	<pre> <xs:element name="THEN"> <xs:annotation> <xs:documentation>Branch taken when booleanResult is True.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </pre>				

element **StepType/DecisionStep/PRECONDITION/THEN/Step**



namespace	http://www.omg.org/space/procspec
type	proc:StepType
properties	isRef 0 minOcc 1 maxOcc unbounded content complex
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>

element StepType/FORstep

diagram	<pre> graph LR A[proc:FORstep] --> B(()) B --> C[proc:LoopVariableName] B --> D[proc:LoopVariableIncrement] B --> E[proc:InitFORStmt] B --> F[proc:TestFORStmt] B --> G[proc:FOR] </pre> <p>Counted loop step.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:LoopVariableName proc:LoopVariableIncrement proc:InitFORStmt proc:TestFORStmt proc:FOR
annotation	documentation Counted loop step.
source	<xs:element name="FORstep"> <xs:annotation> <xs:documentation>Counted loop step.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="LoopVariableName" type="xs:string"> <xs:annotation> <xs:documentation>Name of integer loop variable to be checked. If a Variable array then the

array index is enclosed by one or more brackets () directly after the variable.

```

</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="LoopVariableIncrement" type="xs:integer">
  <xs:annotation>
    <xs:documentation>Value by which the loop variable is incremented (non-zero)</xs:documentation>
  </xs:annotation>
</xs:element>
<xs:element name="InitFORStmt">
  <xs:annotation>
    <xs:documentation>Statement to initialise the loop Variable</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:group ref="proc:StmtHeader"/>
      <xs:group ref="proc:SetVarValue"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="TestFORStmt">
  <xs:annotation>
    <xs:documentation>Statement to check the loop variable. If true the variable is incremented by LoopVariableIncrement and the loop repeated</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:group ref="proc:StmtHeader"/>
      <xs:group ref="proc:CheckVarValue"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="FOR">
  <xs:annotation>
    <xs:documentation>Repeats as long as TestFORStmt is True. stepHeadingLevel = 1 only.</xs:documentation>
  </xs:annotation>
  <xs:complexType>
    <xs:sequence>
      <xs:element name="REPEAT">
        <xs:complexType>
          <xs:sequence>
            <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:sequence>
</xs:complexType>
</xs:element>
```

element StepType/FORstep/LoopVariableName

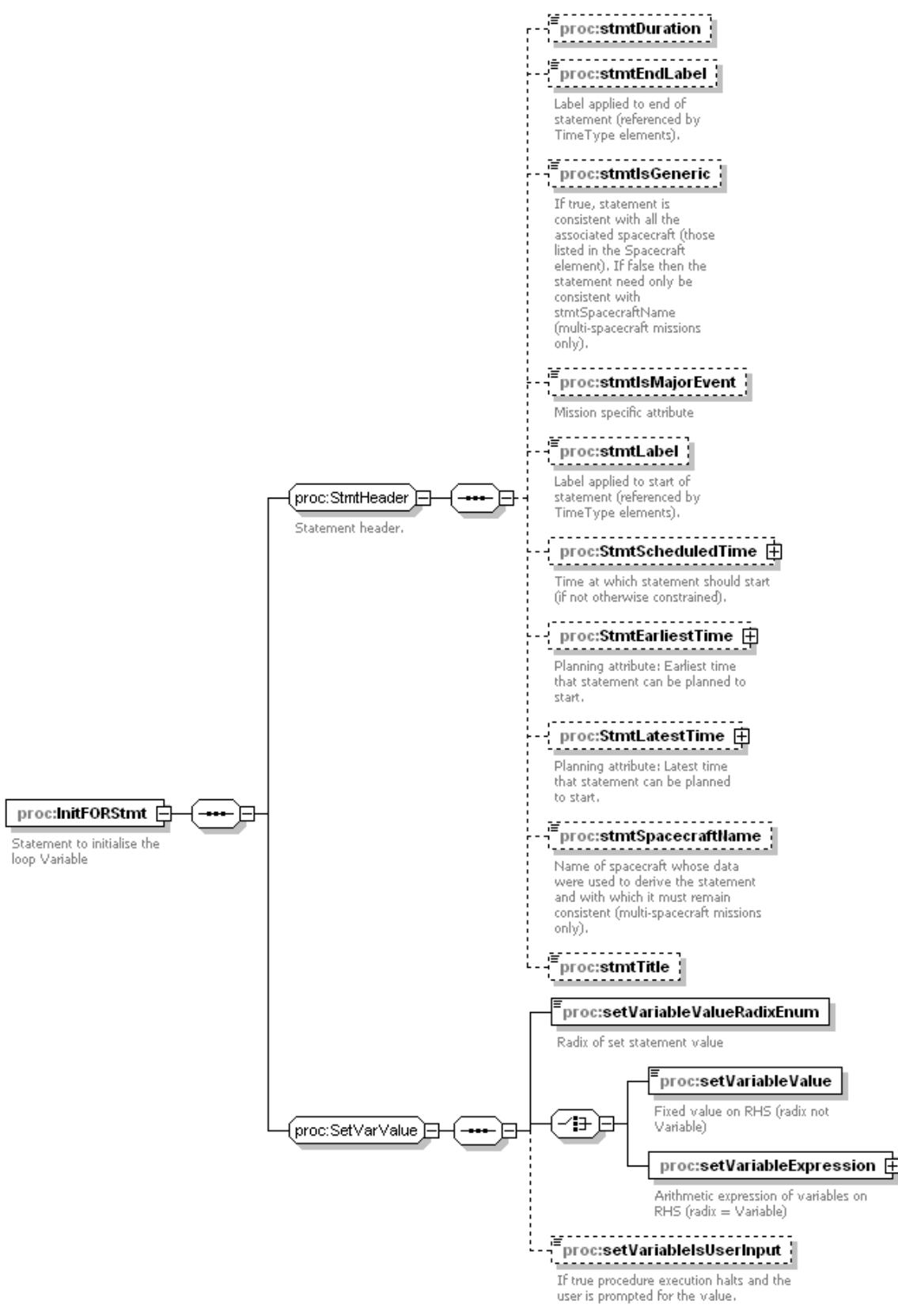
diagram	 proc:LoopVariableName Name of integer loop variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation Name of integer loop variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<xs:element name="LoopVariableName" type="xs:string"> <xs:annotation> <xs:documentation>Name of integer loop variable to be checked. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element>

element StepType/FORstep/LoopVariableIncrement

diagram	 proc:LoopVariableIncrement Value by which the loop variable is incremented (non-zero)
namespace	http://www.omg.org/space/procspec
type	xs:integer
properties	isRef 0 content simple
annotation	documentation Value by which the loop variable is incremented (non-zero)
source	<xs:element name="LoopVariableIncrement" type="xs:integer"> <xs:annotation> <xs:documentation>Value by which the loop variable is incremented (non-zero)</xs:documentation> </xs:annotation> </xs:element>

element StepType/FORstep/InitFORStmt

diagram



namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:setVariableValueRadixEnum proc:setVariableValue proc:setVariableExpression proc:setVariablesUserInput
annotation	documentation Statement to initialise the loop Variable
source	<pre><xs:element name="InitFORStmt"> <xs:annotation> <xs:documentation>Statement to initialise the loop Variable</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:group ref="proc:SetVarValue"/> </xs:sequence> </xs:complexType> </xs:element></pre>

element StepType/FORstep/TestFORstmt

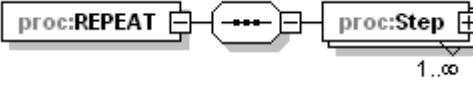
diagram	<pre> classDiagram class proc:TestFORstmt { <<Statement to check the loop variable. If true the variable is incremented by LoopVariableIncrement and the loop repeated>> } class proc:StmtHeader { <<Statement header.>> } class proc:CheckVarValue { <<Value(s) to be checked>> } class proc:stmtDuration class proc:stmtEndLabel class proc:stmtIsGeneric class proc:stmtIsMajorEvent class proc:stmtLabel class proc:stmtScheduledTime class proc:stmtEarliestTime class proc:stmtLatestTime class proc:stmtSpacecraftName class proc:stmtTitle proc:TestFORstmt --> proc:StmtHeader : proc:TestFORstmt --> proc:CheckVarValue : proc:StmtHeader --> proc:stmtDuration : proc:StmtHeader --> proc:stmtEndLabel : proc:StmtHeader --> proc:stmtIsGeneric : proc:StmtHeader --> proc:stmtIsMajorEvent : proc:StmtHeader --> proc:stmtLabel : proc:StmtHeader --> proc:stmtScheduledTime : proc:StmtHeader --> proc:stmtEarliestTime : proc:StmtHeader --> proc:stmtLatestTime : proc:StmtHeader --> proc:stmtSpacecraftName : proc:StmtHeader --> proc:stmtTitle : proc:CheckVarValue --> proc:stmtDuration : proc:CheckVarValue --> proc:stmtEndLabel : proc:CheckVarValue --> proc:stmtIsGeneric : proc:CheckVarValue --> proc:stmtIsMajorEvent : proc:CheckVarValue --> proc:stmtLabel : proc:CheckVarValue --> proc:stmtScheduledTime : proc:CheckVarValue --> proc:stmtEarliestTime : proc:CheckVarValue --> proc:stmtLatestTime : proc:CheckVarValue --> proc:stmtSpacecraftName : proc:CheckVarValue --> proc:stmtTitle : </pre> <p>proc:TestFORstmt Statement to check the loop variable. If true the variable is incremented by LoopVariableIncrement and the loop repeated</p> <p>proc:StmtHeader Statement header.</p> <p>proc:CheckVarValue Value(s) to be checked</p> <p>proc:stmtDuration</p> <p>proc:stmtEndLabel Label applied to end of statement (referenced by TimeType elements).</p> <p>proc:stmtIsGeneric If true, statement is consistent with all the associated spacecraft (those listed in the Spacecraft element). If false then the statement need only be consistent with stmtSpacecraftName (multi-spacecraft missions only).</p> <p>proc:stmtIsMajorEvent Mission specific attribute</p> <p>proc:stmtLabel Label applied to start of statement (referenced by TimeType elements).</p> <p>proc:stmtScheduledTime Time at which statement should start (if not otherwise constrained).</p> <p>proc:stmtEarliestTime Planning attribute: Earliest time that statement can be planned to start.</p> <p>proc:stmtLatestTime Planning attribute: Latest time that statement can be planned to start.</p> <p>proc:stmtSpacecraftName Name of spacecraft whose data were used to derive the statement and with which it must remain consistent (multi-spacecraft missions only).</p> <p>proc:stmtTitle</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:stmtDuration proc:stmtEndLabel proc:stmtIsGeneric proc:stmtIsMajorEvent proc:stmtLabel

	proc:StmtScheduledTime proc:StmtEarliestTime proc:StmtLatestTime proc:stmtSpacecraftName proc:stmtTitle proc:CheckVariableValue
annotation	documentation Statement to check the loop variable. If true the variable is incremented by LoopVariableIncrement and the loop repeated
source	<pre><xs:element name="TestFORStmt"> <xs:annotation> <xs:documentation>Statement to check the loop variable. If true the variable is incremented by LoopVariableIncrement and the loop repeated</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:group ref="proc:StmtHeader"/> <xs:group ref="proc:CheckVarValue"/> </xs:sequence> </xs:complexType> </xs:element></pre>

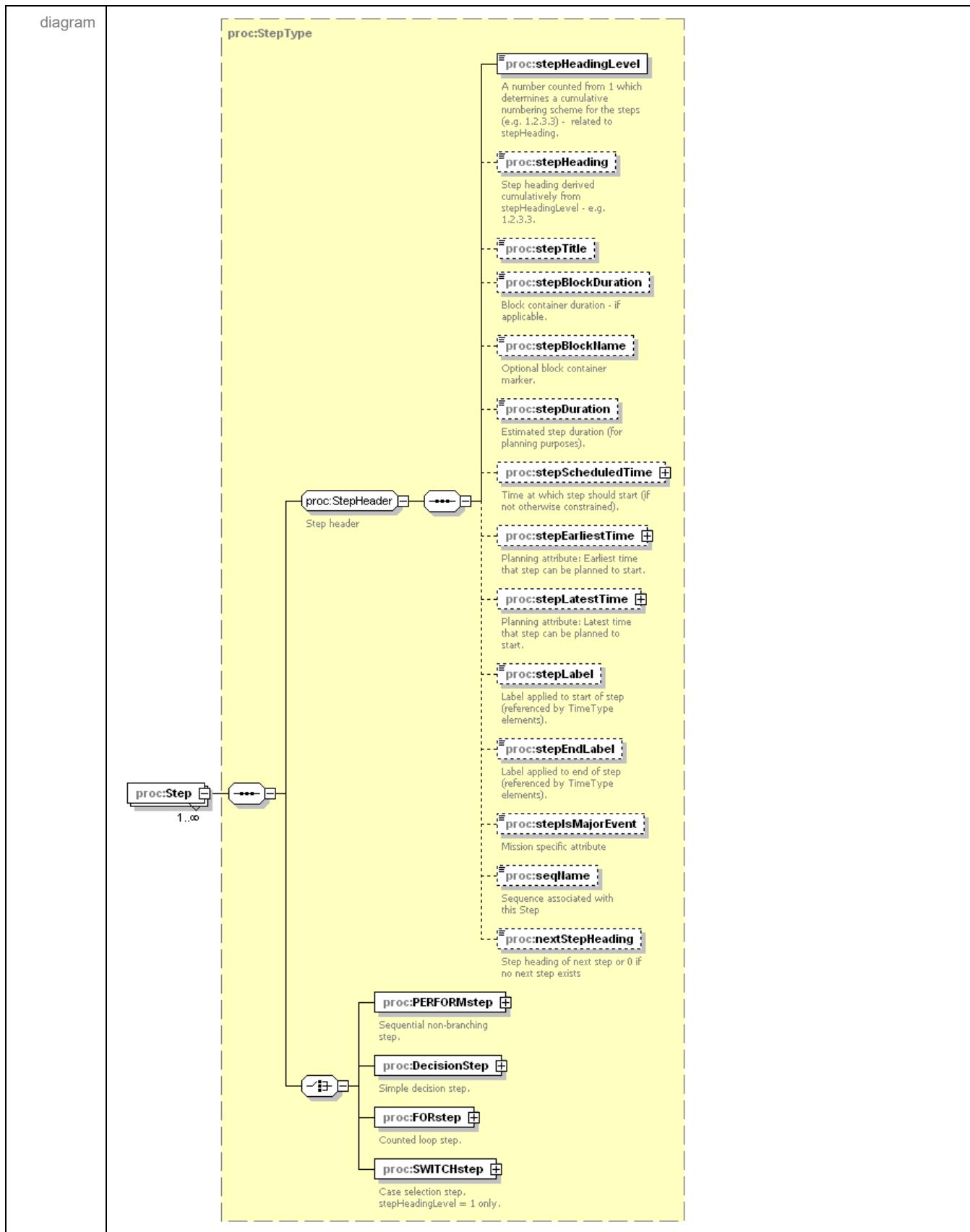
element StepType/FORstep/FOR

diagram	<p>Repeats as long as TestFORStmt is True, stepHeadingLevel = 1 only.</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:REPEAT
annotation	documentation Repeats as long as TestFORStmt is True. stepHeadingLevel = 1 only.
source	<pre><xs:element name="FOR"> <xs:annotation> <xs:documentation>Repeats as long as TestFORStmt is True. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="REPEAT"> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element></pre>

element StepType/FORstep/FOR/REPEAT

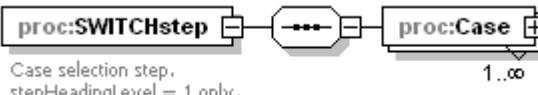
diagram	
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:Step
source	<xs:element name="REPEAT"> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element>

element **StepType/FORstep/FOR/REPEAT/Step**



namespace	http://www.omg.org/space/procspec								
type	proc:StepType								
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>minOcc</td> <td>1</td> </tr> <tr> <td>maxOcc</td> <td>unbounded</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	minOcc	1	maxOcc	unbounded	content	complex
isRef	0								
minOcc	1								
maxOcc	unbounded								
content	complex								
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep								
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>								

element StepType/SWITCHstep

diagram	 <p>Case selection step. stepHeadingLevel = 1 only.</p>				
namespace	http://www.omg.org/space/procspec				
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>complex</td> </tr> </table>	isRef	0	content	complex
isRef	0				
content	complex				
children	proc:Case				
annotation	<p>documentation Case selection step. stepHeadingLevel = 1 only.</p>				
source	<xs:element name="SWITCHstep"> <xs:annotation> <xs:documentation>Case selection step. stepHeadingLevel = 1 only.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Case" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="caseBranchName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of the branch - for display purposes only.</xs:documentation> </xs:annotation> </xs:element> <xs:choice minOccurs="0"> <xs:element name="CaseArithmeticResult"> <xs:annotation> <xs:documentation>The (boolean) result of an arithmetic expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element> <xs:element name="CaseBooleanResult"> <xs:annotation> <xs:documentation>The result of booleanExpression</xs:documentation> </xs:annotation> <xs:complexType>				

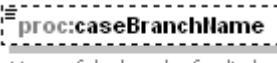
	<pre> <xs:group ref="proc:BooleanResult"/> </xs:complexType> </xs:element> </xs:choice> <xs:element name="CaseExecute"> <xs:annotation> <xs:documentation>Executed if CaseBooleanResult is true.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

element StepType/SWITCHstep/Case

diagram	<pre> classDiagram class proc::Case { <<1..>> <<0..1>> } class proc::caseBranchName { <<Name of the branch - for display purposes only.>> } class proc::CaseArithmeticResult { <<The (boolean) result of an arithmetic expression>> } class proc::CaseBooleanResult { <<The result of booleanExpression>> } class proc::CaseExecute { <<Executed if CaseBooleanResult is true.>> } proc::Case "1..>" -- "*" proc::caseBranchName proc::Case "0..1" -- "*" proc::CaseArithmeticResult proc::Case "0..1" -- "*" proc::CaseBooleanResult proc::Case "0..1" -- "*" proc::CaseExecute </pre>								
namespace	http://www.omg.org/space/procspec								
properties	<table border="1"> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>1</td></tr> <tr> <td>maxOcc</td><td>unbounded</td></tr> <tr> <td>content</td><td>complex</td></tr> </table>	isRef	0	minOcc	1	maxOcc	unbounded	content	complex
isRef	0								
minOcc	1								
maxOcc	unbounded								
content	complex								
children	proc:caseBranchName proc:CaseArithmeticResult proc:CaseBooleanResult proc:CaseExecute								
source	<pre> <xs:element name="Case" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="caseBranchName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of the branch - for display purposes only.</xs:documentation> </xs:annotation> </xs:element> <xs:choice minOccurs="0"> <xs:element name="CaseArithmeticResult"> </pre>								

	<pre> <xs:annotation> <xs:documentation>The (boolean) result of an arithmetic expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element> <xs:element name="CaseBooleanResult"> <xs:annotation> <xs:documentation>The result of booleanExpression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:BooleanResult"/> </xs:complexType> </xs:element> </xs:choice> <xs:element name="CaseExecute"> <xs:annotation> <xs:documentation>Executed if CaseBooleanResult is true.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </pre>
--	--

element StepType/SWITCHstep/Case/caseBranchName

diagram	 <p>Name of the branch - for display purposes only.</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:string								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation</p> <p>Name of the branch - for display purposes only.</p>								
source	<pre> <xs:element name="caseBranchName" type="xs:string" minOccurs="0"> <xs:annotation> <xs:documentation>Name of the branch - for display purposes only.</xs:documentation> </xs:annotation> </xs:element> </pre>								

element StepType/SWITCHstep/Case/CaseArithmeticResult

diagram	
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:unaryOperator proc:Bracket
annotation	documentation The (boolean) result of an arithmetic expression
source	<pre><xs:element name="CaseArithmeticResult"> <xs:annotation> <xs:documentation>The (boolean) result of an arithmetic expression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:ArithmeticResult"/> </xs:complexType> </xs:element></pre>

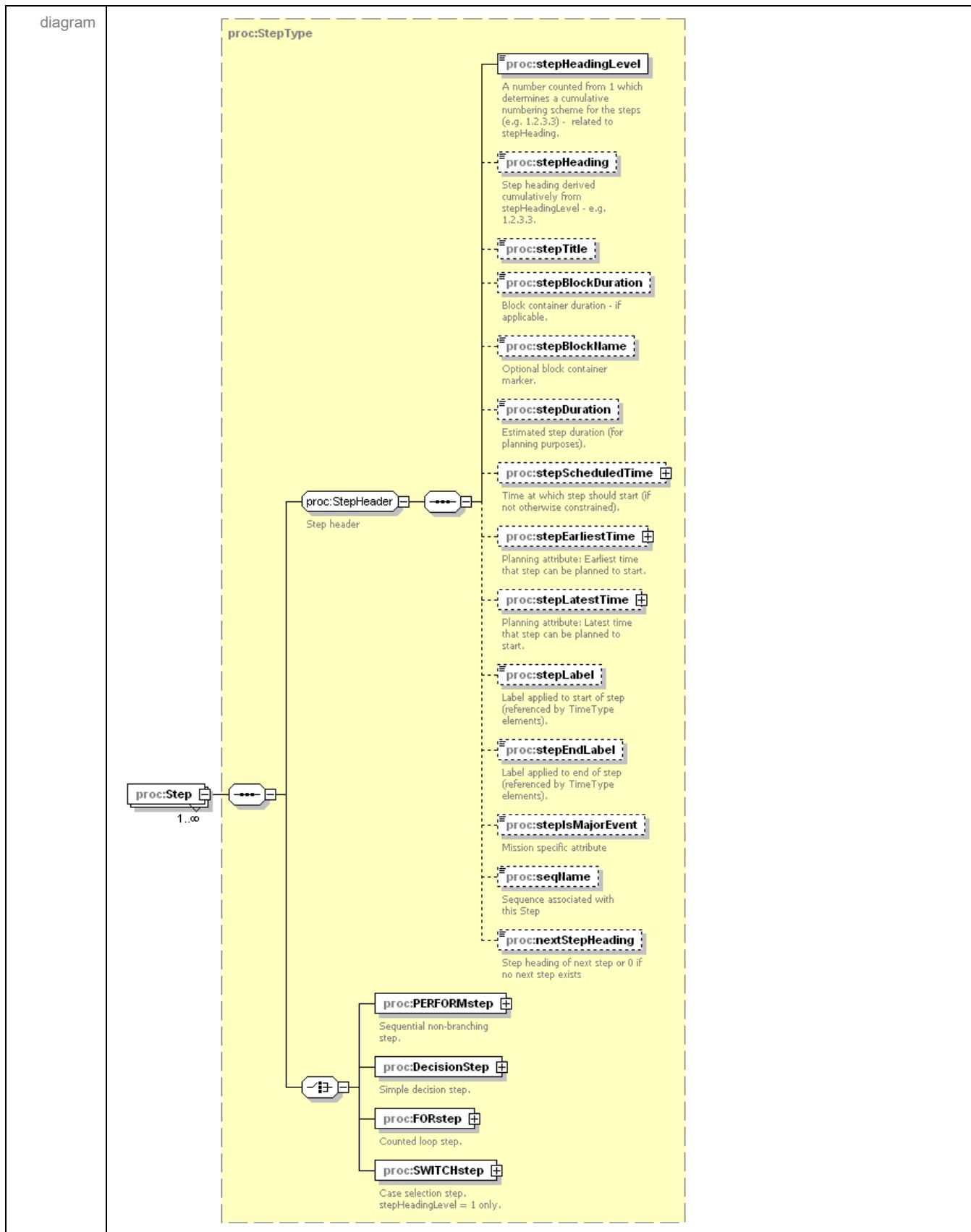
element StepType/SWITCHstep/Case/CaseBooleanResult

diagram	
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:NOT proc:Bracket
annotation	documentation The result of booleanExpression
source	<pre><xs:element name="CaseBooleanResult"> <xs:annotation> <xs:documentation>The result of booleanExpression</xs:documentation> </xs:annotation> <xs:complexType> <xs:group ref="proc:BooleanResult"/> </xs:complexType> </xs:element></pre>

element **StepType/SWITCHstep/Case/CaseExecute**

diagram	<p>Executed if CaseBooleanResult is true.</p> <p>1..∞</p>
namespace	http://www.omg.org/space/procspec
properties	isRef 0 content complex
children	proc:Step
annotation	documentation Executed if CaseBooleanResult is true.
source	<pre><xs:element name="CaseExecute"> <xs:annotation> <xs:documentation>Executed if CaseBooleanResult is true.</xs:documentation> </xs:annotation> <xs:complexType> <xs:sequence> <xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element></pre>

element **StepType/SWITCHstep/Case/CaseExecute/Step**



namespace	http://www.omg.org/space/procspec
type	proc:StepType
properties	isRef 0 minOcc 1 maxOcc unbounded content complex
children	proc:stepHeadingLevel proc:stepHeading proc:stepTitle proc:stepBlockDuration proc:stepBlockName proc:stepDuration proc:stepScheduledTime proc:stepEarliestTime proc:stepLatestTime proc:stepLabel proc:stepEndLabel proc:stepIsMajorEvent proc:seqName proc:nextStepHeading proc:PERFORMstep proc:DecisionStep proc:FORstep proc:SWITCHstep
source	<xs:element name="Step" type="proc:StepType" maxOccurs="unbounded"/>

complexType TimeType

diagram	<p>Absolute time +/- Relative time. Both fields can be a reference to a Variable (of type time). The absolute part can reference a label (defined in a Step or Stmt element).</p> <ul style="list-style-type: none"> proc:absoluteTime: An absolute time literal. proc:absoluteTimeLabel: A label defined in a Step or Statement. proc:absoluteTimeVariable: A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable. proc:negativeSign: A dashed line connects the TimeType element to this element, which then connects to the relativeTime and relativeTimeVariable elements. proc:relativeTime: A relative time literal. proc:relativeTimeVariable: A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
namespace	http://www.omg.org/space/procspec
children	proc:absoluteTime proc:absoluteTimeLabel proc:absoluteTimeVariable proc:negativeSign proc:relativeTime proc:relativeTimeVariable
used by	elements AllStmts/CmdStmt/CmdUplinkTime StepHeader/stepEarliestTime StepHeader/stepLatestTime StepHeader/stepScheduledTime StmtHeader/StmtEarliestTime StmtHeader/StmtLatestTime StmtHeader/StmtScheduledTime complexType ExecutionTimeType
annotation	documentation Absolute time +/- Relative time. Both fields can be a reference to a Variable (of type time). The absolute part can reference a label (defined in a Step or Stmt element).
source	<xs:complexType name="TimeType"> <xs:annotation> <xs:documentation>Absolute time +/- Relative time. Both fields can be a reference to a Variable (of type time). The absolute part can reference a label (defined in a Step or Stmt element).</xs:documentation> </xs:annotation> <xs:sequence>

	<pre> <xs:choice minOccurs="0"> <xs:element name="absoluteTime" type="xs:dateTime"> <xs:annotation> <xs:documentation>An absolute time literal</xs:documentation> </xs:annotation> </xs:element> <xs:element name="absoluteTimeLabel" type="xs:string"> <xs:annotation> <xs:documentation>A label defined in a Step or Statement</xs:documentation> </xs:annotation> </xs:element> <xs:element name="absoluteTimeVariable" type="xs:string"> <xs:annotation> <xs:documentation>A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> </xs:choice> <xs:element name="negativeSign" type="xs:boolean" default="false" minOccurs="0"/> <xs:choice minOccurs="0"> <xs:element name="relativeTime" type="xs:duration"> <xs:annotation> <xs:documentation>A relative time literal</xs:documentation> </xs:annotation> </xs:element> <xs:element name="relativeTimeVariable" type="xs:string"> <xs:annotation> <xs:documentation>A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element> </xs:choice> </xs:sequence> </xs:complexType> </pre>
--	--

element **TimeType/absoluteTime**

diagram	 <p>An absolute time literal</p>
namespace	http://www.omg.org/space/procspec
type	xs:dateTime
properties	isRef 0 content simple
annotation	documentation An absolute time literal
source	<pre> <xs:element name="absoluteTime" type="xs:dateTime"> <xs:annotation> <xs:documentation>An absolute time literal</xs:documentation> </xs:annotation> </xs:element> </pre>

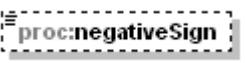
element TimeType/absoluteTimeLabel

diagram	 proc:absoluteTimeLabel A label defined in a Step or Statement
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation A label defined in a Step or Statement
source	<xs:element name="absoluteTimeLabel" type="xs:string"> <xs:annotation> <xs:documentation>A label defined in a Step or Statement</xs:documentation> </xs:annotation> </xs:element>

element TimeType/absoluteTimeVariable

diagram	 proc:absoluteTimeVariable A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<xs:element name="absoluteTimeVariable" type="xs:string"> <xs:annotation> <xs:documentation>A Variable of type ABSOLUTETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element>

element TimeType/negativeSign

diagram	 proc:negativeSign
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple default false
source	<xs:element name="negativeSign" type="xs:boolean" default="false" minOccurs="0"/>

element TimeType/relativeTime

diagram	 A relative time literal
namespace	http://www.omg.org/space/procspec
type	xs:duration
properties	isRef 0 content simple
annotation	documentation A relative time literal
source	<pre><xs:element name="relativeTime" type="xs:duration"> <xs:annotation> <xs:documentation>A relative time literal</xs:documentation> </xs:annotation> </xs:element></pre>

element TimeType/relativeTimeVariable

diagram	 A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.
source	<pre><xs:element name="relativeTimeVariable" type="xs:string"> <xs:annotation> <xs:documentation>A Variable of type RELATIVETIME. If a Variable array then the array index is enclosed by one or more brackets () directly after the variable.</xs:documentation> </xs:annotation> </xs:element></pre>

complexType Variable

diagram	<pre> graph LR Variable[Variable] --- procVariableName[proc:variableName] Variable --- procVariableCategoryEnum[proc:variableCategoryEnum] Variable --- procVariableTypeEnum[proc:variableTypeEnum] Variable --- procVariableDerivedTypeEnum[proc:variableDerivedTypeEnum] Variable --- procVariableProcParamOrder[proc:variableProcParamOrder] Variable --- procVariableDefaultRadixEnum[proc:variableDefaultRadixEnum] Variable --- procVariableDefault[proc:variableDefault] Variable --- procVariableDescription[proc:variableDescription] Variable --- procVariableReadOnly[proc:variableReadOnly] Variable --- procVariableInOut[proc:variableInOut] </pre>
namespace	http://www.omg.org/space/procspec
children	proc:variableName proc:variableCategoryEnum proc:variableTypeEnum proc:variableDerivedTypeEnum proc:variableProcParamOrder proc:variableDefaultRadixEnum proc:variableDefault proc:variableDescription proc:variableReadOnly proc:variableInOut
used by	complexTypes ProcedureVariable SequenceVariable
source	<pre> <xs:complexType name="Variable"> <xs:sequence> <xs:element name="variableName" type="xs:string"> <xs:annotation> </pre>

```

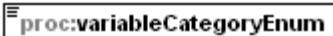
<xs:documentation>The name of the variable</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="variableCategoryEnum">
<xs:annotation>
<xs:documentation>Variables are LOCAL to the procedure, GLOBAL to all procedures or
PROCEDURE PARAMETERS (arguments). There may also be a RETURN VALUE
for
subprocedures.</xs:documentation>
</xs:annotation>
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="LOCAL"/>
<xs:enumeration value="GLOBAL"/>
<xs:enumeration value="PROCEDURE_PARAMETER"/>
<xs:enumeration value="RETURN_VALUE"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="variableTypeEnum" type="proc:variableTypeEnum">
<xs:annotation>
<xs:documentation>If the variable is DERIVED then its exact type is inherited from a TC or
TM parameter - in this case the type is given in the variableDerivedTypeEnum field. Otherwise it
takes a standard variable type (Integer, String etc.). For languages that do not declare their
variables (such as STOL) the type is arbitrary; no type checking will be
performed.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="variableDerivedTypeEnum" type="proc:variableTypeEnum" minOccurs="0">
<xs:annotation>
<xs:documentation>The standard variable type (Integer, String etc.) in case of a DERIVED
variable.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="variableProcParamOrder" type="xs:int" minOccurs="0">
<xs:annotation>
<xs:documentation>Defines an order for the parameter declarations (counted from 1) if
required.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="variableDefaultRadixEnum" type="proc:staticRadixEnum" minOccurs="0"/>
<xs:element name="variableDefault" type="xs:string" minOccurs="0"/>
<xs:element name="variableDescription" type="xs:string" minOccurs="0"/>
<xs:element name="variableReadOnly" type="xs:boolean" default="false" minOccurs="0">
<xs:annotation>
<xs:documentation>True = constant.</xs:documentation>
</xs:annotation>
</xs:element>
<xs:element name="variableInOut" type="xs:boolean" default="false" minOccurs="0">
<xs:annotation>
<xs:documentation>True = in/out parameter, default = in parameter.</xs:documentation>
</xs:annotation>
</xs:element>
</xs:sequence>
</xs:complexType>

```

element Variable/variableName

diagram	 proc:variableName The name of the variable
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 content simple
annotation	documentation The name of the variable
source	<pre><xs:element name="variableName" type="xs:string"> <xs:annotation> <xs:documentation>The name of the variable</xs:documentation> </xs:annotation> </xs:element></pre>

element Variable/variableCategoryEnum

diagram	 proc:variableCategoryEnum Variables are LOCAL to the procedure, GLOBAL to all procedures or PROCEDURE PARAMETERS (arguments). There may also be a RETURN VALUE for subprocedures.
namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
properties	isRef 0 content simple
facets	enumeration LOCAL enumeration GLOBAL enumeration PROCEDURE_PARAMETER enumeration RETURN_VALUE
annotation	documentation Variables are LOCAL to the procedure, GLOBAL to all procedures or PROCEDURE PARAMETERS (arguments). There may also be a RETURN VALUE for subprocedures.
source	<pre><xs:element name="variableCategoryEnum"> <xs:annotation> <xs:documentation>Variables are LOCAL to the procedure, GLOBAL to all procedures or PROCEDURE PARAMETERS (arguments). There may also be a RETURN VALUE for subprocedures.</xs:documentation> </xs:annotation> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="LOCAL"/> <xs:enumeration value="GLOBAL"/> <xs:enumeration value="PROCEDURE_PARAMETER"/> <xs:enumeration value="RETURN_VALUE"/></pre>

	<pre></xs:restriction> </xs:simpleType> </xs:element></pre>
--	---

element Variable/variableTypeEnum

diagram	<p>If the variable is DERIVED then its exact type is inherited from a TC or TM parameter - in this case the type is given in the variableDerivedTypeEnum field. Otherwise it takes a standard variable type (Integer, String etc.). For languages that do not declare their variables (such as STOL) the type is arbitrary; no type checking will be performed.</p>																										
namespace	http://www.omg.org/space/procspec																										
type	proc:variableTypeEnum																										
properties	<table> <tr> <td>isRef</td> <td>0</td> </tr> <tr> <td>content</td> <td>simple</td> </tr> </table>	isRef	0	content	simple																						
isRef	0																										
content	simple																										
facets	<table> <tr><td>enumeration</td><td>DERIVED</td></tr> <tr><td>enumeration</td><td>BOOLEAN</td></tr> <tr><td>enumeration</td><td>BYTE</td></tr> <tr><td>enumeration</td><td>INTEGER</td></tr> <tr><td>enumeration</td><td>LONG_INTEGER</td></tr> <tr><td>enumeration</td><td>UNSIGNED_BYTE</td></tr> <tr><td>enumeration</td><td>UNSIGNED_INTEGER</td></tr> <tr><td>enumeration</td><td>UNSIGNED_LONG_INTEGER</td></tr> <tr><td>enumeration</td><td>REAL</td></tr> <tr><td>enumeration</td><td>DOUBLE</td></tr> <tr><td>enumeration</td><td>RELATIVETIME</td></tr> <tr><td>enumeration</td><td>ABSOLUTETIME</td></tr> <tr><td>enumeration</td><td>STRING</td></tr> </table>	enumeration	DERIVED	enumeration	BOOLEAN	enumeration	BYTE	enumeration	INTEGER	enumeration	LONG_INTEGER	enumeration	UNSIGNED_BYTE	enumeration	UNSIGNED_INTEGER	enumeration	UNSIGNED_LONG_INTEGER	enumeration	REAL	enumeration	DOUBLE	enumeration	RELATIVETIME	enumeration	ABSOLUTETIME	enumeration	STRING
enumeration	DERIVED																										
enumeration	BOOLEAN																										
enumeration	BYTE																										
enumeration	INTEGER																										
enumeration	LONG_INTEGER																										
enumeration	UNSIGNED_BYTE																										
enumeration	UNSIGNED_INTEGER																										
enumeration	UNSIGNED_LONG_INTEGER																										
enumeration	REAL																										
enumeration	DOUBLE																										
enumeration	RELATIVETIME																										
enumeration	ABSOLUTETIME																										
enumeration	STRING																										
annotation	<p>documentation</p> <p>If the variable is DERIVED then its exact type is inherited from a TC or TM parameter - in this case the type is given in the variableDerivedTypeEnum field. Otherwise it takes a standard variable type (Integer, String etc.). For languages that do not declare their variables (such as STOL) the type is arbitrary; no type checking will be performed.</p>																										
source	<pre><xs:element name="variableTypeEnum" type="proc:variableTypeEnum"> <xs:annotation> <xs:documentation>If the variable is DERIVED then its exact type is inherited from a TC or TM parameter - in this case the type is given in the variableDerivedTypeEnum field. Otherwise it takes a standard variable type (Integer, String etc.). For languages that do not declare their variables (such as STOL) the type is arbitrary; no type checking will be performed.</xs:documentation> </xs:annotation> </xs:element></pre>																										

element Variable/variableDerivedTypeEnum

diagram	<p>The standard variable type (Integer, String etc.) in case of a DERIVED variable</p>
namespace	http://www.omg.org/space/procspec

type	<u>proc:variableTypeEnum</u>																										
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple																		
isRef	0																										
minOcc	0																										
maxOcc	1																										
content	simple																										
facets	<table> <tr><td>enumeration</td><td>DERIVED</td></tr> <tr><td>enumeration</td><td>BOOLEAN</td></tr> <tr><td>enumeration</td><td>BYTE</td></tr> <tr><td>enumeration</td><td>INTEGER</td></tr> <tr><td>enumeration</td><td>LONG_INTEGER</td></tr> <tr><td>enumeration</td><td>UNSIGNED_BYTE</td></tr> <tr><td>enumeration</td><td>UNSIGNED_INTEGER</td></tr> <tr><td>enumeration</td><td>UNSIGNED_LONG_INTEGER</td></tr> <tr><td>enumeration</td><td>REAL</td></tr> <tr><td>enumeration</td><td>DOUBLE</td></tr> <tr><td>enumeration</td><td>RELATIVETIME</td></tr> <tr><td>enumeration</td><td>ABSOLUTETIME</td></tr> <tr><td>enumeration</td><td>STRING</td></tr> </table>	enumeration	DERIVED	enumeration	BOOLEAN	enumeration	BYTE	enumeration	INTEGER	enumeration	LONG_INTEGER	enumeration	UNSIGNED_BYTE	enumeration	UNSIGNED_INTEGER	enumeration	UNSIGNED_LONG_INTEGER	enumeration	REAL	enumeration	DOUBLE	enumeration	RELATIVETIME	enumeration	ABSOLUTETIME	enumeration	STRING
enumeration	DERIVED																										
enumeration	BOOLEAN																										
enumeration	BYTE																										
enumeration	INTEGER																										
enumeration	LONG_INTEGER																										
enumeration	UNSIGNED_BYTE																										
enumeration	UNSIGNED_INTEGER																										
enumeration	UNSIGNED_LONG_INTEGER																										
enumeration	REAL																										
enumeration	DOUBLE																										
enumeration	RELATIVETIME																										
enumeration	ABSOLUTETIME																										
enumeration	STRING																										
annotation	<p>documentation The standard variable type (Integer, String etc.) in case of a DERIVED variable</p>																										
source	<pre><xs:element name="variableDerivedTypeEnum" type="proc:variableTypeEnum" minOccurs="0"> <xs:annotation> <xs:documentation>The standard variable type (Integer, String etc.) in case of a DERIVED variable</xs:documentation> </xs:annotation> </xs:element></pre>																										

element Variable/variableProcParamOrder

diagram	<p>proc:variableProcParamOrder</p> <p>Defines an order for the parameter declarations (counted from 1) if required.</p>								
namespace	http://www.omg.org/space/procspec								
type	xs:int								
properties	<table> <tr> <td>isRef</td><td>0</td></tr> <tr> <td>minOcc</td><td>0</td></tr> <tr> <td>maxOcc</td><td>1</td></tr> <tr> <td>content</td><td>simple</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple
isRef	0								
minOcc	0								
maxOcc	1								
content	simple								
annotation	<p>documentation Defines an order for the parameter declarations (counted from 1) if required.</p>								
source	<pre><xs:element name="variableProcParamOrder" type="xs:int" minOccurs="0"> <xs:annotation> <xs:documentation>Defines an order for the parameter declarations (counted from 1) if required.</xs:documentation> </xs:annotation> </xs:element></pre>								

element Variable/variableDefaultRadixEnum

diagram	<p>proc:variableDefaultRadixEnum</p>
namespace	http://www.omg.org/space/procspec
type	<u>proc:staticRadixEnum</u>

properties	isRef 0 minOcc 0 maxOcc 1 content simple
facets	enumeration RAW enumeration RAWHEX enumeration RAWOCT enumeration RAWBIN enumeration CALORALIAS
source	<xs:element name="variableDefaultRadixEnum" type="proc:staticRadixEnum" minOccurs="0"/>

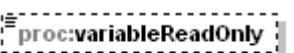
element Variable/variableDefault

diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
source	<xs:element name="variableDefault" type="xs:string" minOccurs="0"/>

element Variable/variableDescription

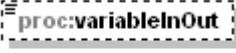
diagram	
namespace	http://www.omg.org/space/procspec
type	xs:string
properties	isRef 0 minOcc 0 maxOcc 1 content simple
source	<xs:element name="variableDescription" type="xs:string" minOccurs="0"/>

element Variable/variableReadOnly

diagram	 True = constant.
namespace	http://www.omg.org/space/procspec
type	xs:boolean
properties	isRef 0 minOcc 0 maxOcc 1 content simple default false
annotation	documentation True = constant.
source	<xs:element name="variableReadOnly" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation>

	<pre><xs:documentation>True = constant.</xs:documentation> </xs:annotation> </xs:element></pre>
--	---

element Variable/variableInOut

diagram	 <p>True = in/out parameter, default = in parameter.</p>										
namespace	http://www.omg.org/space/procspec										
type	xs:boolean										
properties	<table> <tr><td>isRef</td><td>0</td></tr> <tr><td>minOcc</td><td>0</td></tr> <tr><td>maxOcc</td><td>1</td></tr> <tr><td>content</td><td>simple</td></tr> <tr><td>default</td><td>false</td></tr> </table>	isRef	0	minOcc	0	maxOcc	1	content	simple	default	false
isRef	0										
minOcc	0										
maxOcc	1										
content	simple										
default	false										
annotation	<p>documentation True = in/out parameter, default = in parameter.</p>										
source	<pre><xs:element name="variableInOut" type="xs:boolean" default="false" minOccurs="0"> <xs:annotation> <xs:documentation>True = in/out parameter, default = in parameter.</xs:documentation> </xs:annotation> </xs:element></pre>										

simpleType binaryOperator

namespace	http://www.omg.org/space/procspec																																										
type	restriction of xs:string																																										
used by	element ArithmeticResult/Bracket/binaryOperator																																										
facets	<table> <tr><td>enumeration</td><td>+</td></tr> <tr><td>enumeration</td><td>-</td></tr> <tr><td>enumeration</td><td>*</td></tr> <tr><td>enumeration</td><td>/</td></tr> <tr><td>enumeration</td><td>^</td></tr> <tr><td>enumeration</td><td>AND</td></tr> <tr><td>enumeration</td><td>OR</td></tr> <tr><td>enumeration</td><td>XOR</td></tr> <tr><td>enumeration</td><td>==</td></tr> <tr><td>enumeration</td><td>NE</td></tr> <tr><td>enumeration</td><td>LT</td></tr> <tr><td>enumeration</td><td>LE</td></tr> <tr><td>enumeration</td><td>GT</td></tr> <tr><td>enumeration</td><td>GE</td></tr> <tr><td>enumeration</td><td>LEFTSHIFT</td></tr> <tr><td>enumeration</td><td>RIGHTSHIFT</td></tr> <tr><td>enumeration</td><td>BAND</td></tr> <tr><td>enumeration</td><td>BOR</td></tr> <tr><td>enumeration</td><td>BXOR</td></tr> <tr><td>enumeration</td><td>CONCAT</td></tr> <tr><td>enumeration</td><td>MOD</td></tr> </table>	enumeration	+	enumeration	-	enumeration	*	enumeration	/	enumeration	^	enumeration	AND	enumeration	OR	enumeration	XOR	enumeration	==	enumeration	NE	enumeration	LT	enumeration	LE	enumeration	GT	enumeration	GE	enumeration	LEFTSHIFT	enumeration	RIGHTSHIFT	enumeration	BAND	enumeration	BOR	enumeration	BXOR	enumeration	CONCAT	enumeration	MOD
enumeration	+																																										
enumeration	-																																										
enumeration	*																																										
enumeration	/																																										
enumeration	^																																										
enumeration	AND																																										
enumeration	OR																																										
enumeration	XOR																																										
enumeration	==																																										
enumeration	NE																																										
enumeration	LT																																										
enumeration	LE																																										
enumeration	GT																																										
enumeration	GE																																										
enumeration	LEFTSHIFT																																										
enumeration	RIGHTSHIFT																																										
enumeration	BAND																																										
enumeration	BOR																																										
enumeration	BXOR																																										
enumeration	CONCAT																																										
enumeration	MOD																																										
annotation	<p>documentation Binary operators</p>																																										
source	<pre><xs:simpleType name="binaryOperator"> <xs:annotation> <xs:documentation>Binary operators</xs:documentation></pre>																																										

```

</xs:annotation>
<xs:restriction base="xs:string">
  <xs:enumeration value="+"/>
  <xs:enumeration value="-"/>
  <xs:enumeration value="**"/>
  <xs:enumeration value="/">
  <xs:enumeration value="^"/>
  <xs:enumeration value="AND"/>
  <xs:enumeration value="OR"/>
  <xs:enumeration value="XOR"/>
  <xs:enumeration value="=="/>
  <xs:enumeration value="NE"/>
  <xs:enumeration value="LT"/>
  <xs:enumeration value="LE"/>
  <xs:enumeration value="GT"/>
  <xs:enumeration value="GE"/>
  <xs:enumeration value="LEFTSHIFT"/>
  <xs:enumeration value="RIGHTSHIFT"/>
  <xs:enumeration value="BAND"/>
  <xs:enumeration value="BOR"/>
  <xs:enumeration value="BXOR"/>
  <xs:enumeration value="CONCAT"/>
  <xs:enumeration value="MOD"/>
</xs:restriction>
</xs:simpleType>

```

simpleType booleanOperator

namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
used by	element BooleanResult/Bracket/booleanOperator
facets	enumeration AND enumeration OR enumeration XOR
annotation	documentation Standard Boolean operators
source	<xs:simpleType name="booleanOperator"> <xs:annotation> <xs:documentation>Standard Boolean operators</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="AND"/> <xs:enumeration value="OR"/> <xs:enumeration value="XOR"/> </xs:restriction> </xs:simpleType>

simpleType radixEnum

namespace	http://www.omg.org/space/procspec
type	union of (proc:staticRadixEnum , proc:varRadixEnum)
used by	elements CheckVarValue/CheckVariableValue/checkVariableValueRadixEnum Parameter/paramValueRadixEnum AllStmts/SetTelemetryStmt/setTmValueRadixEnum

	SetVarValue/setVariableValueRadixEnum BooleanStmts/TImStmt/TImValueCheck/tImValueRadixEnum
annotation	documentation Dynamic radix types (staticRadixEnum + varRadixEnum).
source	<pre><xs:simpleType name="radixEnum"> <xs:annotation> <xs:documentation>Dynamic radix types (staticRadixEnum + varRadixEnum).</xs:documentation> </xs:annotation> <xs:union memberTypes="proc:staticRadixEnum proc:varRadixEnum"/> </xs:simpleType></pre>

simpleType relationEnum

namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
used by	elements CheckVarValue/CheckVariableValue/checkVariableValueRelationEnum BooleanStmts/TImStmt/TImValueCheck/tImValueRelationEnum
facets	enumeration EQUALS enumeration NOTEQUALS enumeration LESSTHAN enumeration LESSTHANOEQUALS enumeration GREATERTHAN enumeration GREATERTHANOEQUALS
annotation	documentation Standard relation operators
source	<pre><xs:simpleType name="relationEnum"> <xs:annotation> <xs:documentation>Standard relation operators</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="EQUALS"/> <xs:enumeration value="NOTEQUALS"/> <xs:enumeration value="LESSTHAN"/> <xs:enumeration value="LESSTHANOEQUALS"/> <xs:enumeration value="GREATERTHAN"/> <xs:enumeration value="GREATERTHANOEQUALS"/> </xs:restriction> </xs:simpleType></pre>

simpleType staticRadixEnum

namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
used by	element Variable/variableDefaultRadixEnum simpleType radixEnum
facets	enumeration RAW enumeration RAWHEX enumeration RAWOCT enumeration RAWBIN enumeration CALORALIAS enumeration FREETEXT
annotation	documentation Radix representations of parameter values. Numeric raw values are decimal by default (RAW) but can also be input as Hexadecimal, Octal or Binary numbers.
source	<pre><xs:simpleType name="staticRadixEnum"></pre>

	<pre> <xs:annotation> <xs:documentation>Radix representations of parameter values. Numeric raw values are decimal by default (RAW) but can also be input as Hexadecimal, Octal or Binary numbers. If value is neither RAW or CALIBRATED, FREETEXT is used</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="RAW"/> <xs:enumeration value="RAWHEX"/> <xs:enumeration value="RAWOCT"/> <xs:enumeration value="RAWBIN"/> <xs:enumeration value="CALORALIAS"/> <xs:enumeration value="FREETEXT"/> </xs:restriction> </xs:simpleType> </pre>
--	---

simpleType unaryOperator

namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
used by	element ArithmeticResult/unaryOperator
facets	enumeration - enumeration NOT
annotation	documentation Unary arithmetic operators
source	<pre> <xs:simpleType name="unaryOperator"> <xs:annotation> <xs:documentation>Unary arithmetic operators</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="-"/> <xs:enumeration value="NOT"/> </xs:restriction> </xs:simpleType> </pre>

simpleType valueRadixEnum

namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
used by	attribute ArithmeticComponent/fixedValue/@Radix
facets	enumeration Decimal enumeration Hexadecimal enumeration Octal enumeration Binary
annotation	documentation Fixed value unsigned radix types
source	<pre> <xs:simpleType name="valueRadixEnum"> <xs:annotation> <xs:documentation>Fixed value unsigned radix types</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="Decimal"/> <xs:enumeration value="Hexadecimal"/> </pre>

	<pre><xs:enumeration value="Octal"/> <xs:enumeration value="Binary"/> </xs:restriction> </xs:simpleType></pre>
--	--

simpleType variableTypeEnum

namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
used by	elements ProcHeader/ObcmHeader/ObcmParameter/obcpPrmTypeEnum BooleanStmts/TlmStmt/tlmType Variable/variableDerivedTypeEnum Variable/variableTypeEnum
facets	enumeration DERIVED enumeration BOOLEAN enumeration BYTE enumeration INTEGER enumeration LONG_INTEGER enumeration UNSIGNED_BYTE enumeration UNSIGNED_INTEGER enumeration UNSIGNED_LONG_INTEGER enumeration REAL enumeration DOUBLE enumeration RELATIVETIME enumeration ABSOLUTETIME enumeration STRING
annotation	documentation The type of a variable
source	<pre><xs:simpleType name="variableTypeEnum"> <xs:annotation> <xs:documentation>The type of a variable</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="DERIVED"/> <xs:enumeration value="BOOLEAN"/> <xs:enumeration value="BYTE"/> <xs:enumeration value="INTEGER"/> <xs:enumeration value="LONG_INTEGER"/> <xs:enumeration value="UNSIGNED_BYTE"/> <xs:enumeration value="UNSIGNED_INTEGER"/> <xs:enumeration value="UNSIGNED_LONG_INTEGER"/> <xs:enumeration value="REAL"/> <xs:enumeration value="DOUBLE"/> <xs:enumeration value="RELATIVETIME"/> <xs:enumeration value="ABSOLUTETIME"/> <xs:enumeration value="STRING"/> </xs:restriction> </xs:simpleType></pre>

simpleType varRadixEnum

namespace	http://www.omg.org/space/procspec
type	restriction of xs:string
used by	simpleType radixEnum
facets	enumeration TELEMETRYRAWVALUE enumeration TELEMETRYCALVALUE enumeration VARIABLE

	enumeration TELEMTRY enumeration TELECOMMAND enumeration EVENT enumeration OBCP
annotation	documentation Dynamic radix types - variables and telemetry parameter values. VARIABLE means the corresponding value must be an existing Variable entry. TELEMTRY indicates the decimal numeric PID of a telemetry parameter. TELECOMMAND indicates the name of a TC.
source	<pre><xs:simpleType name="varRadixEnum"> <xs:annotation> <xs:documentation>Dynamic radix types - variables and telemetry parameter values. VARIABLE means the corresponding value must be an existing Variable entry. TELEMTRY indicates the decimal numeric PID of a telemetry parameter. TELECOMMAND indicates the name of a TC.</xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:enumeration value="TELEMTRYRAWVALUE"/> <xs:enumeration value="TELEMTRYCALVALUE"/> <xs:enumeration value="VARIABLE"/> <xs:enumeration value="TELEMTRY"/> <xs:enumeration value="TELECOMMAND"/> <xs:enumeration value="EVENT"/> <xs:enumeration value="OBCP"/> </xs:restriction> </xs:simpleType></pre>

APPENDIX 2 DOCUMENT CHANGE RECORD

Issue 1.1 ESOC Additions

Change	Description
varRadixEnum, added enumeration for “TELEMTRY”, “TELECOMMAND”, “EVENT”, OBCP”	Introduced for GAIA OBCP MOIS customisation, n/a for non-OBCP procedures
variableTypeEnum - added “UNSIGNED_BYTE”, “UNSIGNED_INTEGER”, “UNSIGNED_LONG_INTEGER”	Introduced for GAIA OBCP MOIS customisation, n/a for non-OBCP procedures
ExecutionTimeType, new complex type	Requested by Orbit Angle missions, it extends the TimeType
StepHeader added nextStepHeading	
ObcpHeader added obcpName, obcpCode, pidName, pidCode, pidNumber	Introduced for GAIA OBCP MOIS customisation, n/a for non-OBCP procedures
ArithmeticComponent added telemetry	To make a telemetry check part of the arithmetic expression
Parameter added paramDescr,	The last 3 tags are added for GAIA OBCP MOIS

paramEngUnit, paramRawValue, ParamValueParam	customisation, n/a for non-OBCP procedures
TlmStmt added optionals: tlmPid, tlmType, tlmOffset, tlmLength, tlmDescr, tlmEngUnit, tlmGlobalLength	Introduced for GAIA OBCP MOIS customisation, n/a for non-OBCP procedures
PacketStmt added packetDescr	
CmdStmt added cmdDescr, changed CmdExecutionTime	CmdExecutionTime now is of type ExecutionTimeType
ProcCallStmt added procCallDescr	
SeqCallStmt Added seqCallDescr	

Issue 1.2 Industry Update

Change	Description
Variable added variableDerivedTypeEnum	For Astrium Sentinel-2 translator. Provide source type for variables of type “DERIVED” (SPR6324)
Parameter added groupRepeatSize	Required for Astrium ELISA export (SPR6290)
TlmStmt added tlmDetailedDescr	Required for Astrium BepiColombo TCL processor (SPR6277)
CmdStmt added cmdDetailedDescr, SubScheduleId	Required for Astrium BepiColombo TCL processor (SPR6277)
Spacecraft added databaseVersion group and generationDate	For Astrium Sentinel-2 translator (SPR6283)

Issue 1.3 Industry Update (DLR)

Change	Description
Empty Perform steps allowed (p 220, 227,228)	DLR (SPR6522)