
**Information technology — Document
Schema Definition Languages (DSDL) —**

**Part 4:
Namespace-based Validation Dispatching
Language (NVDL)**

iTeh STANDARD PREVIEW
(standards.itih.ai)

*Technologies de l'information — Langages de définition de schéma de
documents (DSDL)*

*Partie 4: Langage de diffusion de validation d'espace de nom orienté
(NVDL) EC 19757-4:2006*

[https://standards.itih.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-
1f929271b945/iso-iec-19757-4-2006](https://standards.itih.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1f929271b945/iso-iec-19757-4-2006)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 19757-4:2006

<https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1f929271b945/iso-iec-19757-4-2006>

© ISO/IEC 2006

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword.....	v
Introduction.....	vi
1 Scope.....	1
2 Normative references.....	1
3 Terms and definitions.....	2
4 Notation.....	3
5 Data model.....	3
5.1 General.....	3
5.2 Creating a data model from the infoSet.....	4
6 Syntax.....	6
6.1 General.....	6
6.2 Full syntax.....	6
6.3 Simple syntax.....	8
6.4 Simplification.....	10
6.4.1 General.....	10
6.4.2 Annotations.....	10
6.4.3 Whitespace.....	10
6.4.4 message attribute.....	10
6.4.5 mustSupport attribute.....	10
6.4.6 schemaType attribute of rules elements.....	11
6.4.7 rules without mode children.....	11
6.4.8 child mode elements of validate, allow, reject, attach, unwrap, attachPlaceholder, or context elements..	11
6.4.9 namespace or anyNamespace elements.....	11
6.4.10 mode inclusion.....	11
6.4.11 competition within mode	12
6.4.12 default anyNamespace	12
6.4.13 allow and reject	12
6.4.14 useMode attribute.....	13
7 Primitive operations.....	13
7.1 General.....	13
7.2 Creating element sections and attribute sections.....	13
7.3 Decomposition of element sections by trigger elements.....	15
7.4 Attaching attribute sections to elements.....	17
7.5 Attaching element sequences to elements.....	17
7.6 Creating placeholder elements from element sections.....	18
7.7 Converting attribute sections to empty elements.....	18
8 Semantics.....	18
8.1 General.....	18
8.2 Preliminaries.....	19
8.3 Stage 1: Creating element and attribute sections.....	20
8.4 Stage 2: Constructing interpretations.....	20
8.5 Stage 3: Combining sections.....	20
8.6 Stage 4: Filtering of the combined sections.....	22
8.7 Stage 5: Validation.....	22
8.7.1 General.....	22
8.7.2 Determining schemas and schema languages.....	22
8.7.3 Schema rewriting for attribute sections.....	23
8.7.4 Options.....	23
9 Conformance.....	23

Annex A (normative) Full syntax in RELAX NG.....	24
Annex B (normative) Simple syntax in RELAX NG.....	31
Annex C (informative) An NVDL script and RELAX NG schema for the full syntax.....	35
C.1 General.....	35
C.2 RELAX NG schema.....	35
C.3 NVDL script.....	40
Annex D (informative) Example.....	42
D.1 General.....	42
D.2 RDF embedded within XHTML.....	42
D.2.1 Simplification.....	42
D.2.2 Dispatching.....	43
D.2.2.1 General.....	43
D.2.2.2 Stage 1.....	43
D.2.2.3 Stage 2.....	44
D.2.2.4 Stage 3.....	44
D.2.2.5 Stage 4.....	44
D.2.2.6 Stage 5.....	44
D.3 XHTML 2.0 and XForms.....	44
D.3.1 Simplification.....	44
D.3.2 Dispatching.....	47
D.3.2.1 General.....	47
D.3.2.2 Stage 1.....	47
D.3.2.3 Stage 2.....	47
D.3.2.4 Stage 3.....	48
D.3.2.5 Stage 4.....	48
D.3.2.6 Stage 5.....	49
Bibliography.....	50

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 19757-4:2006
<https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1f929271b945/iso-iec-19757-4-2006>

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 19757-4 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 34, *Document description and processing languages*.

ISO/IEC 19757 consists of the following parts, under the general title *Information technology — Document Schema Definition Languages (DSDL)*:

- *Part 1: Overview* <https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1f929271b945/iso-iec-19757-4-2006>
- *Part 2: Regular-grammar-based validation — RELAX NG*
- *Part 3: Rule-based validation — Schematron*
- *Part 4: Namespace-based Validation Dispatching Language (NVDL)*

The following parts are under preparation:

- *Part 5: Datatypes*
- *Part 6: Path-based integrity constraints*
- *Part 7: Character repertoire description language — CRDL*
- *Part 8: Document schema renaming language — DSRL*
- *Part 9: Datatype- and namespace-aware DTDs*
- *Part 10: Validation management*

Introduction

ISO/IEC 19757 defines a set of Document Schema Definition Languages (DSDL) that can be used to specify one or more validation processes performed against Extensible Markup Language (XML) documents. A number of validation technologies are standardized in DSDL to complement those already available as standards or from industry.

The main objective of ISO/IEC 19757 is to bring together different validation-related technologies to form a single extensible framework that allows technologies to work in series or in parallel to produce a single or a set of validation results. The extensibility of DSDL accommodates validation technologies not yet designed or specified.

The motivations of this part of ISO/IEC 19757 are twofold. One is to allow the interworking of schemas describing different markup vocabularies. The other is to allow these schemas to be written in different schema languages. For this purpose, this part of ISO/IEC 19757 specifies a Namespace-based Validation Dispatching Language (NVDL).

The structure of this part of ISO/IEC 19757 is as follows. Clause 5 describes the data model, which is the abstraction of an XML document used throughout the rest of the document. Clause 6 describes the full syntax and the simple syntax of NVDL scripts, and further describes the transformation from the full syntax to the simple syntax. Clause 7 describes primitive operations for the NVDL data model, which are used for defining the NVDL semantics. Clause 8 describes the semantics of a correct NVDL script in the simple syntax; the semantics specify how elements and attributes in a given document are dispatched to different validators and which schema is used by each of these validators. Clause 9 describes conformance requirements for NVDL dispatchers. Annex A and Annex B define the full syntax and the simple syntax using RELAX NG, respectively. Annex C defines the full syntax using NVDL and RELAX NG. Finally, Annex D provides examples of the application of NVDL.

The origin of NVDL is JIS TR X 0044[4], which was created and then submitted to ISO/IEC JTC1 as a fast-track ISO/IEC DTR 22250-2[5] by the Japanese national member body for SC 34.

ISO/IEC 19757-4:2006
<https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1f929271b945/iso-iec-19757-4-2006>

Information technology — Document Schema Definition Languages (DSDL) —

Part 4: Namespace-based Validation Dispatching Language (NVDL)

1 Scope

This part of ISO/IEC 19757 specifies a Namespace-based Validation Dispatching Language (NVDL). An NVDL script controls the dispatching of elements or attributes in a given XML document to different validators, depending on the namespaces of the elements or attributes. An NVDL script also specifies which schemas are used by these validators. These schemas may be written in any schema languages, including those specified by ISO/IEC 19757.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

NOTE Each of the following documents has a unique identifier that is used to cite the document in the text. The unique identifier consists of the part of the reference up to the first comma.

ISO/IEC 19757-2, *Information technology — Document Schema Definition Languages (DSDL) — Part 2: Regular-grammar-based validation — RELAX NG*

W3C XML, *Extensible Markup Language (XML) 1.0 (Third Edition)*, W3C Recommendation, 04 February 2004, available at <http://www.w3.org/TR/2004/REC-xml-20040204/>

W3C XML-Names, *Namespaces in XML*, W3C Recommendation, 14 January 1999, available at <http://www.w3.org/TR/1999/REC-xml-names-19990114/>

W3C XML-Infoset, *XML Information Set (Second Edition)*, W3C Recommendation, 4 February 2004, available at <http://www.w3.org/TR/2004/REC-xml-infoset-20040204/>

W3C XML Schema Part 2, *XML Schema Part 2: Datatypes (Second Edition)*, W3C Recommendation, 28 October 2004, available at <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/>

IETF RFC 2045, *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies*, Internet Standards Track Specification, November 1996, available at <http://www.ietf.org/rfc/rfc2045.txt>

IETF RFC 2046, *Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types*, Internet Standards Track Specification, November 1996, available at <http://www.ietf.org/rfc/rfc2046.txt>

IETF RFC 3023, *XML Media Types*, Internet Standards Track Specification, August 1998, available at <http://www.ietf.org/rfc/rfc3023.txt>

IETF RFC 3986, *Uniform Resource Identifiers (URI): Generic Syntax*, Internet Standards Track Specification, January 2005, available at <http://www.ietf.org/rfc/rfc3986.txt>

IETF RFC 3987, *Internationalized Resource Identifiers (IRIs)*, Internet Standards Track Specification, January 2005, available at <http://www.ietf.org/rfc/rfc3987.txt>

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19757-2 and the following apply.

NOTE Where a term is defined in this part of ISO/IEC 19757 and in ISO/IEC 19757-2, the definition given in this part of ISO/IEC 19757 applies.

- 3.1
action**
a validate, reject, allow, attach, unwrap, cancelNestedActions, or attachPlaceholder element within an NVDL script
- 3.2
attribute section**
a non-empty set of attributes having the same namespace name
- 3.3
attribute slot node**
a slot for an attribute section
- 3.4
NVDL dispatcher**
software module that determines whether an NVDL script is correct, creates validation candidates from an instance, and invokes validators for these validation candidates
- 3.5
element section**
an element such that a single namespace name applies to itself and all descendant elements
- 3.6
element slot node**
a slot for an element section
- 3.7
full syntax**
syntax of an NVDL script before simplification
- 3.8
instance**
XML document from which validation candidates are created
- 3.9
media type**
a two-part identifier specifying the nature of the referenced data
- 3.10
mode**
a mapping from namespaces to actions
- 3.11
NVDL data model**
abstract representation of an XML document defined by this part of ISO/IEC 19757
- 3.12
NVDL script**
specification for namespace-based validation dispatching
- 3.13
parent element section**
an element section containing a slot node for another section
- 3.14
path expression**
a list of one or more choices separated by |, where each choice is a list of one or more NCName tokens separated by /, optionally preceded by /

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO/IEC 19757-4:2006](https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1029271b945/iso-iec-19757-4-2006)

<https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1029271b945/iso-iec-19757-4-2006>

3.15**RELAX NG data model**

abstract representation of an XML document defined by ISO/IEC 19757-2

3.16**section**

either an attribute section or element section

3.17**simple syntax**

syntax of an NVDL script after simplification

3.18**simplification**

transformation of an NVDL script in the full syntax to a script in the simple syntax

3.19**slot node**

either an attribute slot node or element slot node

3.20**validation candidate**

an element not having any slot nodes as descendants

NOTE Different elements in a validation candidate may belong to different namespaces. Different attributes of an element in a validation candidate may belong to different namespaces.

4 Notation

iTeh STANDARD PREVIEW
(standards.iteh.ai)

The following notations have been adopted in the formal definitions provided in this part of ISO/IEC 19757.

< [..] >: the context of an element in the NVDL data model

< {..} >: the namespace name of an element in the NVDL data model

(...): a sequence of elements

{...}: a set of attributes

x+y: the concatenation of two element sequences x and y

e <- [sn, x]: an element constructed from e by replacing a slot node sn with an element sequence or attribute section x

|e|: the number of descendant elements in an element e

createPlaceholder(s): a placeholder element created from an element section s as specified in 7.6.

virtualElement(s): an empty element created from an attribute section s as specified in 7.7

path(s): the path from the parent element section of s to s as defined in 8.2

5 Data model**5.1 General**

NVDL deals with XML documents representing both schemas and instances through an abstract data model. XML documents representing schemas and instances shall be well-formed in conformance with W3C XML and shall conform to the constraints of W3C XML-Names.

This abstract NVDL data model is an extension of the RELAX NG data model. Names, contexts, attributes, and strings in the NVDL data model are identical to those in the RELAX NG data model defined in ISO/IEC 19757-2. Elements in the NVDL data model are extended as described in this clause.

An element consists of:

- a name,
- a context,
- a set of attributes or attribute slot nodes, and
- an ordered sequence of zero or more children; each child is either an element, a non-empty string, or an element slot node; the sequence never contains two consecutive strings.

NOTE 1 Since contexts include mappings from prefixes to namespace names, validators invoked by NVDL dispatchers can handle qualified names occurring in attribute values or element contents.

An element (say *e*) is said to be a descendant of another element (say *e'*) when *e* occurs in the child sequence of either *e'* or some descendant element of *e'*. Meanwhile, *e* is said to be an ancestor of *e'* when *e'* is a descendant of *e*.

An attribute slot node is a slot for a non-empty set of attributes. An attribute slot node is said to belong to an element (say *e*) when this attribute slot node is contained by the set of attributes or attribute slot nodes of either *e* or some descendant element of *e*.

An element slot node is a slot for an element. An element slot node is said to belong to an element (say *e*) when this element slot node occurs in the child sequence of either *e* or some descendant element of *e*.

NOTE 2 An element (say *e*) is not a descendant of another element (say *e'*) if *e* has an element slot node for *e*.

NOTE 3 Attribute or element slot nodes do not exist in the RELAX NG data model.

5.2 Creating a data model from the infoset

This process is identical to the process defined in ISO/IEC 19757-2.

EXAMPLE 1 Consider an XML document as follows.

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:foo xmlns:ns1="http://www.example.com/one" xmlns:ns2="http://www.example.com/two">
<!-- comment-example -->
<?pi-example?>
  <ns1:foo1>
    text1
  </ns1:foo1>
  <ns2:foo2>
    text2
  </ns2:foo2>
  <ns1:foo3>
    text3
  </ns1:foo3>
</ns1:foo>
```

We use an XML-like syntax for representing data models, but there are two changes. First, in each start tag, the tag name is replaced by a namespace name, local name, and context. Second, each end tag is represented by "</>".

The data model created from this document is shown below. (Whitespace is slightly changed for readability.) Every element in this data model references to a context (say cx1) that maps ns1 to "http://www.example.com/one" and ns2 to "http://www.example.com/two", respectively.

```
<{http://www.example.com/one}foo [cx1]>
  <{http://www.example.com/one}foo1 [cx1]>
    text1
    <{http://www.example.com/two}foo11 [cx1]></>
  </>
  <{http://www.example.com/two}foo2 [cx1]>
    text2
    <{http://www.example.com/one}foo21 [cx1]></>
    text3
    <{http://www.example.com/one}foo22 [cx1]></>
  </>
  <{http://www.example.com/one}foo3 [cx1]>
    <{http://www.example.com/one}foo31 [cx1]></>
  </>
</>
```

Observe that the XML declaration and any comments or processing instructions in the input XML document do not appear in the data model and that there are no empty-element tags.

EXAMPLE 2 The following XML document is obtained by adding attributes to the document in the previous example.

```
<?xml version="1.0" encoding="UTF-8"?>
<ns1:foo xmlns:ns1="http://www.example.com/one" xmlns:ns2="http://www.example.com/two">
<!-- comment-example -->
<?pi-example?>
  <ns1:foo1 bar1="_" bar2="_">
    text1
    <ns2:foo11/>
  </ns1:foo1>
  <ns2:foo2 ns1:bar1="_" ns1:bar2="_" ns2:bar1="_" ns2:bar2="_" bar1="_" bar2="_">
    text2
    <ns1:foo21/>
    text3
    <ns1:foo22/>
  </ns2:foo2>
  <ns1:foo3 ns2:bar1="_" ns2:bar2="_">
    <ns1:foo31/>
  </ns1:foo3>
</ns1:foo>
```

The data model created from this document is shown below. cx1 is the same as in Example 1. Each attribute name is replaced by a namespace name and local name.

```
<{http://www.example.com/one}foo [cx1]>
  <{http://www.example.com/one}foo1 [cx1]
    {bar1="_" bar2="_"}
    text1
    <{http://www.example.com/two}foo11 [cx1]></>
  </>
  <{http://www.example.com/two}foo2 [cx1]
    {http://www.example.com/one}bar1="_"
    {http://www.example.com/one}bar2="_"
    {http://www.example.com/two}bar1="_"
    {http://www.example.com/two}bar2="_"
    {bar1="_" bar2="_"}
  </>
```

```

text2
<{http://www.example.com/one}foo21 [cx1]></>
text3
<{http://www.example.com/one}foo22 [cx1]></>
</>
<{http://www.example.com/one}foo3 [cx1]
  {http://www.example.com/two}bar1="_"
  {http://www.example.com/two}bar2="_">
  <{http://www.example.com/one}foo31 [cx1]></>
</>
</>

```

6 Syntax

6.1 General

NVDL has the full syntax and the simple syntax. A correct NVDL script is always required to be in the full syntax described in 6.2. Meanwhile, the simple syntax described in 6.3 is purely an internal artifact for specifying the semantics of NVDL. NVDL scripts in the full syntax may be transformed to NVDL scripts in the simple syntax.

6.2 Full syntax

An NVDL script in the full syntax shall be an XML document valid against the following RELAX NG schema in the compact syntax. The same schema in the XML syntax is shown in Annex A.

```

namespace local = ""
default namespace nvd1 = "http://purl.oclc.org/dsdl/nvd1/ns/structure/1.0"

start =
  element rules {
    (schemaType?,
     trigger*,
     (rule* | (attribute startMode { xsd:NCName }, mode+)))
    & foreign
  }

trigger =
  element trigger {
    (attribute ns { xsd:string },
     attribute nameList { list { xsd:NCName } })
    & foreign
  }

mode =
  element mode {
    (attribute name { xsd:NCName },
     includedMode*,
     rule*)
    & foreign
  }

includedMode =
  element mode {
    (attribute name { xsd:NCName }?,
     includedMode*,
     rule*)
    & foreign
  }

```

```

rule =
  element namespace {
    (attribute ns { xsd:string },
    attribute wildcard {xsd:string{maxLength = "1"}}?,
    ruleModel)
    & foreign
  }
| element anyNamespace { ruleModel & foreign}

ruleModel = attribute match { elementsOrAttributes }?, actions

elementsOrAttributes =
  list {
    ("elements", "attributes")
    | ("attributes", "elements")
    | "elements"
    | "attributes"
  }

actions =
  cancelAction |
  (noResultAction*, (noResultAction|resultAction), noResultAction*)

cancelAction =
  element cancelNestedActions {foreign}

noResultAction =
  element validate {
    (schemaType?,
    (message | option)*,
    schema,
    modeUsage) & foreign
  }
| element allow|reject { (message*, modeUsage) & foreign }

schema =
  attribute schema { xsd:anyURI } |
  element schema {(text | foreignElement), foreignAttribute*}

message =
  attribute message {text}
  | element message {(text, xmlAttribute*), nonXMLForeignAttribute*}

resultAction =
  element attach|attachPlaceholder|unwrap { (message*, modeUsage) & foreign }

option =
  element option {
    (attribute name { xsd:anyURI },
    attribute arg { text }?,
    attribute mustSupport { xsd:boolean }?)
    & foreign
  }

modeUsage =
  (attribute useMode { xsd:NCName }
  | nestedMode)?,
  element context {
    (attribute path { path },
    (attribute useMode { xsd:NCName }
    | nestedMode)?
    ) & foreign
  }*

nestedMode =
  element mode {

```

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 19757-4:2006
<https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-1f929271b945/iso-iec-19757-4-2006>

```

    (includedMode*,
    rule*)
    & foreign
  }

schemaType = attribute schemaType { mediaType }

## 5.1 of RFC 2045 allows <any (US-ASCII) CHAR except SPACE, CTLs,
## or tspecials>, where
##
##    tspecials := "(" / ")" / "<" / ">" / "@" /
##               "," / ";" / ":" / "\" / "<" / ">"
##               "/" / "[" / "]" / "?" / "="
##
mediaType =
  xsd:string {
    pattern = "\s*"
      ~ "[0-9A-Za-z!#$%&'\+|-.\|_`{\|}~]*"
      ~ "/"
      ~ "[0-9A-Za-z!#$%&'\+|-.\|_`{\|}~]*"
      ~ "\s*"
  }

path =
  xsd:string {
    pattern = "\s*(\/s*)?i\c*(s*\/s*i\c*)*s*"
      ~ "(\/s*(\/s*)?i\c*(s*\/s*i\c*)*s*)"
  }

foreignElement =
  element (* - nvdl:*) { attribute * {text}*, mixed {anyElement*}}

anyElement =
  element * { attribute * {text}*, mixed {anyElement*}}

foreignAttribute = attribute * - (nvdl:* | local:*) {text}

nonXMLForeignAttribute = attribute * - (xml:* | nvdl:* | local:*) {text}

xmlAttribute =
  attribute xml:lang {text}
  | attribute xml:space {"default" | "preserve"}
  | attribute xml:base {xsd:anyURI}

foreign = foreignAttribute*, foreignElement*

```

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 19757-4:2006
<https://standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-45-iso-iec-19757-4-2006>

6.3 Simple syntax

The simple syntax is a subset of the full syntax. An NVDL script in the simple syntax shall be an XML document valid against the following RELAX NG schema in the compact syntax. The same schema in the XML syntax is shown in Annex B.

```

namespace local = ""
default namespace nvdl = "http://purl.oclc.org/dsdl/nvdl/ns/structure/1.0"

start =
  element rules {
    attribute startMode { xsd:NCName }, trigger*, mode+
  }

trigger =
  element trigger {
    attribute ns { xsd:string },

```

```

    attribute nameList { list { xsd:NCName } }
  }

mode =
  element mode {
    attribute name { xsd:NCName },
    rule*
  }

rule =
  element namespace {
    attribute ns { xsd:string },
    attribute wildCard { xsd:string{maxLength = "1"}},
    ruleModel
  }
| element anyNamespace { ruleModel }

ruleModel = attribute match { elementsOrAttributes }, actions

elementsOrAttributes =
  "elements" | "attributes"

actions =
  (noResultAction*, (noResultAction|resultAction), noResultAction*)

noResultAction =
  element validate {
    schemaType?,
    (message | option)*,
    schema,
    modeUsage
  }

schema =
  attribute schema { xsd:anyURI },
  element schema {(text | foreignElement)}

message =
  element message {text & attribute xml:lang {text}}

resultAction =
  element attach|attachPlaceholder|unwrap { message*, modeUsage }

option =
  element option {
    attribute name { xsd:anyURI },
    attribute arg { text }?,
    attribute mustSupport { xsd:boolean }
  }

modeUsage =
  attribute useMode { xsd:NCName },
  element context {
    attribute path { path },
    attribute useMode { xsd:NCName }
  }*

schemaType = attribute schemaType { mediaType }

## 5.1 of RFC 2045 allows <any (US-ASCII) CHAR except SPACE, CTLs,
## or tspecials>, where
##
##   tspecials := "(" / ")" / "<" / ">" / "@" /
##              "," / ";" / ":" / "\" / "<" / ">"
##              "/" / "[" / "]" / "?" / "="
##

```

iTeh STANDARD PREVIEW
(standards.iteh.ai)

ISO/IEC 19757-4:2006

standards.iteh.ai/catalog/standards/sist/f91459f0-d05e-4297-be24-29271b945/iso-iec-19757-4-2006