



Guide to Schematron Rules and Patterns

SRC Schematron Guide

Version 2015-AUGr2022-MAY

December 1, 2022

Distribution Notice:

This document has been approved for Public Release and is available for use without restriction.

Table of Contents

Chapter 1 - Introduction	1
1.1 - Purpose	1
1.2 - Overview	1
1.3 - Schematron	1
1.4 - Conformance	1
Chapter 2 - Rules	2
2.1 - ../Rules/generalConstraints/SRC_ID_00001.sch	3
2.2 - ../Rules/TDFConstraints/SRC_ID_00002.sch	4
2.3 - ../Rules/TDFConstraints/SRC_ID_00003.sch	5
2.4 - ../Rules/TDFConstraints/SRC_ID_00004.sch	6
2.5 - ../Rules/timeZoneIndicators/SRC_ID_00006.sch	7
2.6 - ../Rules/datesAndTimes/SRC_ID_00007.sch	10
2.7 - ../Rules/nonNullConstraints/SRC_ID_00009.sch	11
2.8 - ../Rules/elementBasedContraints/SRC_ID_00010.sch	14
2.9 - ../Rules/generalConstraints/SRC_ID_00011.sch	15
2.10 - ../Rules/generalConstraints/SRC_ID_00012.sch	16
2.11 - ../Rules/generalConstraints/SRC_ID_00013.sch	17
2.12 - ../Rules/generalConstraints/SRC_ID_00014.sch	18
2.13 - ../Rules/generalConstraints/SRC_ID_00015.sch	19
2.14 - ../Rules/generalConstraints/SRC_ID_00016.sch	20
2.15 - ../Rules/generalConstraints/SRC_ID_00017.sch	21
2.16 - ../Rules/generalConstraints/SRC_ID_00018.sch	22
2.17 - ../Rules/generalConstraints/SRC_ID_00019.sch	23
2.18 - ../Rules/generalConstraints/SRC_ID_00020.sch	24
Chapter 3 - Abstract Patterns	25
3.1 - ../Lib/ValidateValidationEnvCVE.sch	26
3.2 - ../Lib/ValidateValidationEnvSchema.sch	27
Chapter 4 - Schematron Schema	28
4.1 - ../SRC_XML.sch	29
Chapter 5 - Removed Rules	31
5.1 - ../Rules/deleted/SRC_ID_00005.sch	31
5.2 - ../Rules/deleted/SRC_ID_00008.sch	31

Chapter 1 - Introduction

1.1 - Purpose

This is an informative supplement for SRC. This guide is generated from the SRC Schematron rules and provides a consolidated reference for the business rules of this specification.

1.2 - Overview

Chapter 2 is a listing of all the numbered rules in SRC. For each rule, there is a rule description, a code description, and a code block with the Schematron rule.

Chapter 3 is a listing of abstract patterns used in SRC. The abstract patterns may be used in numbered rules or provided as reference for use in rules developed by users of SRC. Each abstract pattern has a code description and a code block with the abstract Schematron pattern.

Chapter 4 is a listing of the master SRC Schematron file with all of the imports of rules and patterns. Many of the rules and patterns listed in Chapters 3 and 4 rely on functions and variables defined in the master file.

Chapter 5 is a listing of rules that have been deleted.

1.3 - Schematron

The business rules for SRC are encoded using ISO Schematron. Schematron is a rule-based validation language that uses XML Path Language to make assertions about an XML document.

SRC uses the XSLT 2.0 implementation of Schematron by Rick Jelliffe (2010-04-14) as its reference implementation. The only available identifying descriptors for this implementation are the implementer's name and date of release. This implementation may be found at the following URL: <http://code.google.com/p/schematron/>.



Important

The Schematron rules in this specification use XSLT 2.0 query binding.

1.4 - Conformance

This guide is informative. The Schematron rules listed here are normative in the sense that they convey criteria that a document **MUST** adhere to, exactly as English may be used to convey normative criteria. It is not necessary for implementers to use the specific Schematron encoding in this specification. Implementers **MAY** use any encodings, tools, or languages desired to implement validation schemes for conformance to this specification. However, to conform to the specification, validation schemes **MUST** match the behavior of the reference Schematron implementation. That is, a validator **MUST** find a document valid *if and only if* the reference Schematron implementation would find the document valid according to SRC's Schematron rules.

Chapter 2 - Rules

All of the numbered Rules for SRC are listed in this section. These rules may depend on patterns defined in the Abstract Patterns section or on variables defined in the Schematron Schema section.

Rules identifiers are all of the format SRC-ID-XXXXX, with rule files named SRC_ID_XXXXX.sch. Any other heading indicates a supporting file that may influence a rule but is not actually a numbered rule.

2.1 - ../Rules/generalConstraints/SRC_ID_00001.sch

Rule Description

[SRC-ID-00001][Error] Regardless of the version indicated on the instance document, the validation infrastructure MUST use a version of 'ISM' that is version '202111' (Version:2021-NOV) or later. NOTE: This is not an error of the instance document but of the validation environment itself.

Code Description

This rule uses an abstract pattern to consolidate logic. It verifies that the validation infrastructure is using the version specified in parameters.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00001" is-a="ValidateValidationEnvCVE">
    <sch:param name="MinVersion" value="'202111'"/>
    <sch:param name="SpecToCheck" value="'ISM'"/>
    <sch:param name="pathToDocument"
        value="'../../CVE/ISM/CVEnumISMClassificationAll.xml'"/>
    <sch:param name="RuleID" value="'SRC-ID-00001'"/>
</sch:pattern>
```

2.2 - ../Rules/TDFConstraints/SRC_ID_00002.sch

Rule Description

[SRC-ID-00002][Error] The SourceID values within a TDO are unique. Human Readable: The SourceID values within a TDO are unique.

Code Description

For each tdf:TrustedDataObject, check to make sure that the total number of src:SourceID/src:DocumentID and src:SourceID/ucid:Identifier values equal the total number of distinct src:SourceID/src:DocumentID and src:SourceID/ucid:Identifier values.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00002">
    <sch:rule id="SRC-ID-00002-R1" context="tdf:TrustedDataObject">
        <sch:assert test="(count(tdf:Assertion/tdf:StructuredStatement/src:SourceCitations//src:SourceID/src:DocumentID) + count(tdf:Assertion/tdf:StructuredStatement/
src:SourceCitations//src:SourceID/ucid:Identifier)) = (count(distinct-values(tdf:Assertion/tdf:StructuredStatement/src:SourceCitations//src:SourceID/src:DocumentID)) + count(distinct-
values(tdf:Assertion/tdf:StructuredStatement/src:SourceCitations//src:SourceID/ucid:Identifier)))"
            flag="error"
            role="error">[SRC-ID-00002][Error] The SourceID values within a TDO are unique. Human Readable: The SourceID values within a TDO are unique.</sch:assert>
        </sch:rule>
    </sch:pattern>
```

2.3 - ../Rules/TDFConstraints/SRC_ID_00003.sch

Rule Description

[SRC-ID-00003][Error] src:SourceCitations element must have ancestor that is a tdf:TrustedDataObject. Human Readable: SRC assertions must live within a TDO.

Code Description

The ancestor of each src:SourceCitations element must be some tdf:TrustedDataObject.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00003">
    <sch:rule id="SRC-ID-00003-R1" context="src:SourceCitations">
        <sch:assert test="ancestor::tdf:TrustedDataObject" flag="error" role="error">[SRC-ID-00003][Error] src:SourceCitations element must have ancestor that is a
tdf:TrustedDataObject. Human Readable: SRC assertions must live within a TDO.</sch:assert>
    </sch:rule>
</sch:pattern>
```


2.4 - ../Rules/TDFConstraints/SRC_ID_00004.sch

Rule Description

[SRC-ID-00004][Error] Each TDO can have at most 1 src:SourceCitations element. Human Readable: There can only be one Source Citation Root Element per TDO.

Code Description

For each tdf:TrustedDataObject, this rule ensures that the count of src:SourceCitations element lesser than or equal to 1.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00004">
    <sch:rule id="SRC-ID-00004-R1" context="tdf:TrustedDataObject">
        <sch:assert test="count(child::tdf:Assertion/tdf:StructuredStatement/src:SourceCitations)&lt;= 1"
                    flag="error"
                    role="error">[SRC-ID-00004][Error] Each TDO can have at most 1 src:SourceCitations element. Human Readable: There can only be one Source Citation Root
Element per TDO.</sch:assert>
    </sch:rule>
</sch:pattern>
```

2.5 - `./Rules/timeZoneIndicators/SRC_ID_00006.sch`

Rule Description

[SRC-ID-00006][Warning] For elements `ApproximableDateTime`, `DatePublished`, `DateTimeReferenced`, `DateString`, `EarliestStartDate`, `LatestEndDate`, if the time designator (T) is specified, it is recommended that time zone be specified. Human Readable: It is recommended that time zone be specified if time designator (T) is specified for the following elements: `ApproximableDateTime`, `DatePublished`, `DateTimeReferenced`, `DateString`, `EarliestStartDate`, `LatestEndDate`.

Code Description

This pattern uses an abstract rule to consolidate logic. If the value of the context contains the time zone designator (T), then it makes sure that the value of the context matches the regular expression for a date with a time zone specified. The abstract rule is extended once for each required element listed in rule SRC-ID-00006.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00006"><!-- Abstract rule for elements, which asserts that if the time designator (T) is specified, then the timezone is specified -->

<sch:rule abstract="true" id="abs.rule00006">
    <sch:assert test="if (contains(string($valueTest),'T')) then matches(string($valueTest),$endsWithTimeZoneRegEx) else true()"
        flag="warning"
        role="warning">[SRC-ID-00006][Warning] For element
        <sch:name/>, if the time designator (T) is specified, it is recommended that time zone be specified. Human Readable: It is recommended that time zone be specified if time designator (T)
is specified for the following elements: ApproximableDateTime, DatePublished, DateTimeReferenced, DateString, EarliestStartDate, LatestEndDate.
    </sch:assert>
</sch:rule>
<!-- Abstract rule for attributes, which asserts that if the time designator (T) is specified, then the timezone is specified -->

<sch:rule abstract="true" id="abs.rule00006attrs">
    <sch:let name="fails"
        value="for $date in $dateValues return if(if (contains(string($date),'T')) then matches(string($date),$endsWithTimeZoneRegEx) else true()) then null else
name($date)"/>
    <sch:assert test="count($fails)=0" flag="warning" role="warning">[SRC-ID-00006][Warning] For attribute(s):
    <sch:value-of select="for $each in $fails return concat(' ', string($each))"/>if the time designator (T) is specified, it is recommended that the time zone be specified.
    </sch:assert>
</sch:rule>
<!-- Begin using abstract rule on required elements -->

<sch:rule id="SRC-ID-00006-R3" context="src:ApproximableDateTime">
    <sch:let name="valueTest" value="."/>
    <sch:extends rule="abs.rule00006"/>
</sch:rule>
<sch:rule id="SRC-ID-00006-R4" context="src:DatePublished">
    <sch:let name="valueTest" value="."/>
    <sch:extends rule="abs.rule00006"/>
</sch:rule>
<sch:rule id="SRC-ID-00006-R5" context="src:DateTimeReferenced">
    <sch:let name="valueTest" value="."/>
    <sch:extends rule="abs.rule00006"/>
</sch:rule>
<sch:rule id="SRC-ID-00006-R6" context="src:DateString">
    <sch:let name="valueTest" value="."/>
    <sch:extends rule="abs.rule00006"/>
</sch:rule>
<sch:rule id="SRC-ID-00006-R7" context="src:EarliestStartDate">
    <sch:let name="valueTest" value="."/>
    <sch:extends rule="abs.rule00006"/>
</sch:rule>
<sch:rule id="SRC-ID-00006-R8" context="src:LatestEndDate">
    <sch:let name="valueTest" value="."/>
    <sch:extends rule="abs.rule00006"/>
</sch:rule>
```

</sch:pattern>

2.6 - ../Rules/datesAndTimes/SRC_ID_00007.sch

Rule Description

[SRC-ID-00007][Error] If the DateInformation element exists, at least one of its child elements DateString, ApproximableDateTime, or SearchableDateTime must be present. Human Readable: Element DateInformation must have a value for one of its child elements.

Code Description

For element src:DateInformation, this rule ensures that one or more of the child elements DateString, ApproximableDateTime, SearchableDateTime/EarliestStartDate or SearchableDateTime/LatestEndDate is specified with a non-white space value.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00007">
    <sch:rule id="SRC-ID-00007-R1" context="src:DateInformation">
        <sch:assert test="normalize-space(string(src:DateString)) or normalize-space(string(src:ApproximableDateTime)) or normalize-space(string(src:SearchableDateTime/
src:EarliestStartDate)) or normalize-space(string(src:SearchableDateTime/src:LatestEndDate))"
            flag="error"
            role="error">[SRC-ID-00007][Error] If the DateInformation element exists, at least one of its child elements DateString, ApproximableDateTime, or
SearchableDateTime must be present. Human Readable: Element DateInformation must have a value for one of its child elements.</sch:assert>
        </sch:rule>
    </sch:pattern>
```

2.7 - `./Rules/nonNullConstraints/SRC_ID_00009.sch`

Rule Description

[SRC-ID-00009][Warning] For every optional element that exists in the document and can have text content, the element should have non-null, non-whitespace value.

Code Description

This pattern uses an abstract rule to consolidate logic. The abstract rule first concatenates the text values within the given element, separated by a single space. The resultant string is then normalized with leading and trailing whitespace removed, and the length of the string is determined to be greater than zero, which indicates non-whitespace content. The abstract rule is extended once for each optional element in the SRC schema.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00009">
    <sch:rule abstract="true" id="abs.rule00009">
        <sch:assert test="normalize-space(string())" flag="warning" role="warning">[SRC-ID-00009][Warning] For every optional element that exists in the document and can have
text content, the element should have non-null, non-whitespace value.</sch:assert>
    </sch:rule>
    <!-- Begin using abstract rule on optional elements -->

<sch:rule id="SRC-ID-00009-R2" context="src:InformationOriginator">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R3" context="src:ARCList">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R4" context="src:SourceSummaryStatement">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R5" context="src:AlternateSourceID">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R6" context="src:SourceDescriptor">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R7" context="src:Language">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R8" context="src:DatePublished">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R9" context="src:DateTimeReferenced">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R10" context="src:DateInformation">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R11" context="src:Title">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R12" context="src:SubTitle">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R13" context="src:CompilationTitle">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R14" context="src:EditionNumber">
    <sch:extends rule="abs.rule00009"/>
</sch:rule>
```

```
<sch:rule id="SRC-ID-00009-R15" context="src:VolumeNumber">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R16" context="src:IssueNumber">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R17" context="src:SegmentReferenced">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R18" context="src:Link">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R19" context="src:InclusionReason">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R20" context="src:Agency">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R21" context="src:AuthorInfo">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R22" context="src:EditorInfo">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R23" context="src:POCinfo">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R24" context="src:OriginalClassificationMarking">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R25" context="src:DateString">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R26" context="src:ApproximableDateTime">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R27" context="src:SearchableDateTime">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
<sch:rule id="SRC-ID-00009-R28" context="src:ContactFor">
  <sch:extends rule="abs.rule00009"/>
</sch:rule>
</sch:pattern>
```


2.8 - ../Rules/elementBasedConstraints/SRC_ID_00010.sch

Rule Description

[SRC-ID-00010][Error] For elements: PersonalProfileGroup at least one of the following child elements must have non-whitespace content: Surname, UserID, JobTitle, Affiliation, OfficeName, PostalAddress, PhoneNumber, FaxNumber, EmailAddress, WebPageAddress, FormattedSignatureBlock. Human Readable: PersonalProfileGroup must have a value for at least one of the following child elements: Surname, UserID, JobTitle, Affiliation, OfficeName, PostalAddress, PhoneNumber, FaxNumber, EmailAddress, WebPageAddress, FormattedSignatureBlock.

Code Description

This pattern uses an abstract rule to consolidate logic. It normalizes the space of the value of the specified child elements and makes sure that the length of the resulting string is greater than zero, which indicates non-whitespace content. Element PostalAddress cannot contain text content, so the rule counts the number of its child elements that contain non-white space and makes sure that the count is great than 0.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00010"><!-- Abstract rule, which asserts that at least one of the listed child elements has non-whitespace content -->

<sch:rule abstract="true" id="abs.rule00010">
    <sch:assert test="src:Surname[normalize-space(string(text()))] or src:UserID[normalize-space(string(text()))] or src:JobTitle[normalize-space(string(text()))] or
src:Affiliation[normalize-space(string(text()))] or src:OfficeName[normalize-space(string(text()))] or src:PhoneNumber[normalize-space(string(text()))] or src:FaxNumber[normalize-
space(string(text()))] or src:EmailAddress[normalize-space(string(text()))] or src:WebPageAddress[normalize-space(string(text()))] or src:FormattedSignatureBlock[normalize-
space(string(text()))] or (some $token in src:PostalAddress/*/text() satisfies normalize-space(string($token)))"
        flag="error"
        role="error">[SRC-ID-00010][Error] For element
    <sch:name/>at least one of the following child elements must have non-whitespace content: Surname, UserID, JobTitle, Affiliation, OfficeName, PostalAddress, PhoneNumber, FaxNumber,
EmailAddress, WebPageAddress, FormattedSignatureBlock. Human Readable: Elements PersonalProfileGroup must have a value for at least one of the following child elements: Surname, UserID,
JobTitle, Affiliation, OfficeName, PostalAddress, PhoneNumber, FaxNumber, EmailAddress, WebPageAddress, FormattedSignatureBlock.
    </sch:assert>
</sch:rule>
<!-- Begin using abstract rule to check required elements -->

<sch:rule id="SRC-ID-00010-R2" context="src:AuthorInfo">
    <sch:extends rule="abs.rule00010"/>
</sch:rule>
<sch:rule id="SRC-ID-00010-R3" context="src:POCinfo">
    <sch:extends rule="abs.rule00010"/>
</sch:rule>
</sch:pattern>
```

2.9 - ../Rules/generalConstraints/SRC_ID_00011.sch

Rule Description

[SRC-ID-00011][Error] For SRC elements with INTDIS attributes or containing INTDIS elements, the INTDIS CESVersion must exist somewhere in the instance. Human Readable: If SRC is using INTDIS in any way, the INTDIS CESVersion must exist in the instance.

Code Description

For SRC elements with INTDIS attributes or containing INTDIS elements, the INTDIS CESVersion must exist somewhere in the instance.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00011">
    <sch:rule id="SRC-ID-00011-R1" context="//src:*[@intdis:* or ../intdis:*)>
        <sch:assert test="//*[@intdis:CESVersion]" flag="error" role="error">[SRC-ID-00011][Error] For SRC elements with INTDIS attributes or containing INTDIS elements, the
INTDIS CESVersion must exist somewhere in the instance. Human Readable: If SRC is using INTDIS in any way, the INTDIS CESVersion must exist in the instance.</sch:assert>
    </sch:rule>
</sch:pattern>
```

2.10 - ../Rules/generalConstraints/SRC_ID_00012.sch

Rule Description

[SRC-ID-00012][Error] Regardless of the version indicated on the instance document, the validation infrastructure MUST use a version of 'INTDIS' that is version '201707' (Version:2017-JUL) or later. NOTE: This is not an error of the instance document but of the validation environment itself.

Code Description

This rule uses an abstract pattern to consolidate logic. It verifies that the validation infrastructure is using the version specified in parameters.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00012" is-a="ValidateValidationEnvCVE">
    <sch:param name="MinVersion" value="'201707'"/>
    <sch:param name="SpecToCheck" value="'INTDIS'"/>
    <sch:param name="pathToDocument"
        value="'../../../../CVE/INTDIS/CVEnumIntelDiscipline.xml'"/>
    <sch:param name="RuleID" value="'SRC-ID-00012'"/>
</sch:pattern>
```

2.11 - ../Rules/generalConstraints/SRC_ID_00013.sch

Rule Description

[SRC-ID-00013][Error] Regardless of the version indicated on the instance document, the validation infrastructure MUST use a version of 'VIRT' that is version '202010' (Version:2020-OCT) or later. NOTE: This is not an error of the instance document but of the validation environment itself.

Code Description

This rule uses an abstract pattern to consolidate logic. It verifies that the validation infrastructure is using the version specified in parameters.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00013" is-a="ValidateValidationEnvSchema">
    <sch:param name="MinVersion" value="'202010'"/>
    <sch:param name="SpecToCheck" value="'VIRT'"/>
    <sch:param name="pathToDocument" value="'../Schema/VIRT/VIRT.xsd'"/>
    <sch:param name="RuleID" value="'SRC-ID-00013'"/>
</sch:pattern>
```

2.12 - ../Rules/generalConstraints/SRC_ID_00014.sch

Rule Description

[SRC-ID-00014][Error] Regardless of the version indicated on the instance document, the validation infrastructure MUST use a version of 'IC-ID' that is version '1' or later. NOTE: This is not an error of the instance document but of the validation environment itself.

Code Description

This rule uses an abstract pattern to consolidate logic. It verifies that the validation infrastructure is using the version specified in parameters.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00014" is-a="ValidateValidationEnvSchema">
    <sch:param name="MinVersion" value="'1'"/>
    <sch:param name="SpecToCheck" value="'IC-ID'"/>
    <sch:param name="pathToDocument" value="'../Schema/IC-ID/IC-ID.xsd'"/>
    <sch:param name="RuleID" value="'SRC-ID-00014'"/>
</sch:pattern>
```

2.13 - ../Rules/generalConstraints/SRC_ID_00015.sch

Rule Description

[SRC-ID-00015][Error] Regardless of the version indicated on the instance document, the validation infrastructure MUST use a version of 'IC-EDH' that is version '201903' (Version:2019-MAR) or later. NOTE: This is not an error of the instance document but of the validation environment itself.

Code Description

This rule uses an abstract pattern to consolidate logic. It verifies that the validation infrastructure is using the version specified in parameters.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00015" is-a="ValidateValidationEnvSchema">
    <sch:param name="MinVersion" value="'201903'"/>
    <sch:param name="SpecToCheck" value="'IC-EDH'"/>
    <sch:param name="pathToDocument" value="'../Schema/IC-EDH/IC-EDH.xsd'"/>
    <sch:param name="RuleID" value="'SRC-ID-00015'"/>
</sch:pattern>
```

2.14 - ../Rules/generalConstraints/SRC_ID_00016.sch

Rule Description

[SRC-ID-00016][Error] For SRC elements with IC-EDH attributes or containing IC-EDH elements, the IC-EDH DESVersion must exist somewhere in the instance. Human Readable: If SRC is using IC-EDH in any way, the IC-EDH DESVersion must exist in the instance.

Code Description

For SRC elements with IC-EDH attributes or containing IC-EDH elements, the IC-EDH DESVersion must exist somewhere in the instance.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00016">
    <sch:rule id="SRC-ID-00016-R1" context="//src:*[@edh:* or ../edh:*]">
        <sch:assert test="//*[@edh:DESVersion]" flag="error" role="error">[SRC-ID-00016][Error] For SRC elements with IC-EDH attributes or containing IC-EDH elements, the IC-EDH DESVersion must exist somewhere in the instance. Human Readable: If SRC is using IC-EDH in any way, the IC-EDH DESVersion must exist in the instance.</sch:assert>
        </sch:rule>
    </sch:pattern>
```

2.15 - ../Rules/generalConstraints/SRC_ID_00017.sch

Rule Description

[SRC-ID-00017][Error] For SRC elements with VIRT attributes or containing VIRT elements, the VIRT DESVersion must exist somewhere in the instance. Human Readable: If SRC is using VIRT in any way, the VIRT DESVersion must exist in the instance.

Code Description

For SRC elements with VIRT attributes or containing VIRT elements, the VIRT DESVersion must exist somewhere in the instance.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00017">
    <sch:rule id="SRC-ID-00017-R1" context="//src:*[@virt:* or ../virt:*]>
        <sch:assert test="//*[@virt:DESVersion]" flag="error" role="error">[SRC-ID-00017][Error] For SRC elements with VIRT attributes or containing VIRT elements, the VIRT
DESVersion must exist somewhere in the instance. Human Readable: If SRC is using VIRT in any way, the VIRT DESVersion must exist in the instance.</sch:assert>
        </sch:rule>
    </sch:pattern>
```


2.16 - ../Rules/generalConstraints/SRC_ID_00018.sch

Rule Description

[SRC-ID-00018][Error] For SRC elements with IC-ID attributes or containing IC-ID elements, the IC-ID DESVersion must exist somewhere in the instance. Human Readable: If SRC is using IC-ID in any way, the IC-ID DESVersion must exist in the instance.

Code Description

For SRC elements with IC-ID attributes or containing IC-ID elements, the IC-ID DESVersion must exist somewhere in the instance.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00018">
    <sch:rule id="SRC-ID-00018-R1" context="//src:*[@icid:* or ../icid:*)>
        <sch:assert test="//*[@icid:DESVersion]" flag="error" role="error">[SRC-ID-00018][Error] For SRC elements with IC-ID attributes or containing IC-ID elements, the IC-ID
DESVersion must exist somewhere in the instance. Human Readable: If SRC is using IC-ID in any way, the IC-ID DESVersion must exist in the instance.</sch:assert>
    </sch:rule>
</sch:pattern>
```

2.17 - ../Rules/generalConstraints/SRC_ID_00019.sch

Rule Description

[SRC-ID-00019][Error] For SRC elements with ISM attributes or containing ISM elements, the ISM DESVersion must exist somewhere in the instance. Human Readable: If SRC is using ISM in any way, the ISM DESVersion must exist in the instance.

Code Description

For SRC elements with ISM attributes or containing ISM elements, the ISM DESVersion must exist somewhere in the instance.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00019">
    <sch:rule id="SRC-ID-00019-R1" context="//src:*[@ism:* or ../ism:*)>
        <sch:assert test="//*[@ism:DESVersion]" flag="error" role="error">[SRC-ID-00019][Error] For SRC elements with ISM attributes or containing ISM elements, the ISM
DESVersion must exist somewhere in the instance. Human Readable: If SRC is using ISM in any way, the ISM DESVersion must exist in the instance.</sch:assert>
        </sch:rule>
    </sch:pattern>
```

2.18 - .//Rules/generalConstraints/SRC_ID_00020.sch

Rule Description

[SRC-ID-00020][Warning] src:DESVersion attribute SHOULD be specified as version 201508.202205 (Version:2015-AUG Revision: 2022-MAY) with an optional extension.

Code Description

This rule supports extending the version identifier with an optional trailing hyphen and up to 23 additional characters. The version must match the regular expression “^201508.202205(-.{1,23})?\$”.

Schematron Code

```
<?ICEA pattern?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:pattern id="SRC-ID-00020">
    <sch:rule id="SRC-ID-00020-R1" context="*[@src:DESVersion]">
        <sch:assert test="matches(@src:DESVersion, '^201508.202205(-.{1,23})?$')"
            flag="warning"
            role="warning">[SRC-ID-00020][Warning] src:DESVersion attribute SHOULD be specified as version 201508.202205 (Version:2015-AUG Revision: 2022-MAY) with an
optional extension.</sch:assert>
        </sch:rule>
    </sch:pattern>
```

Chapter 3 - Abstract Patterns

All of the Abstract Patterns for SRC are listed in this section. These patterns may depend on variables defined in the Schematron Schema section.

3.1 - ./Lib/ValidateValidationEnvCVE.sch

Code Description

This abstract pattern checks to see if the validation environment has at least the version / revision of the CVE as of the writing of this specification. The calling rule must pass in \$MinVersion, \$SpecToCheck, \$pathToDocument, \$RuleID.

Schematron Code

```
<!--
  This abstract pattern checks to see the version of a CVE is greater than or equal to a passed in parameter.

  $MinVersion      := the version that SpecToCheck must be equal to or greater than.
  $SpecToCheck      := Name the spec whose version in the infrastructure is being checked.
  $pathToDocument   := Relative path to the document cve that has ther version string
  $RuleID           := The number of the rule in the concrete file.
-->

<sch:pattern xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
              abstract="true"
              id="ValidateValidationEnvCVE">
  <sch:rule id="ValidateValidationEnvCVE-R1" context="/">
    <sch:assert test="document($pathToDocument)//cve:CVE//@specVersion castable as xs:double and document($pathToDocument)//cve:CVE//@specVersion >= $MinVersion"
               flag="error"
               role="error">[
  <sch:value-of select="$RuleID"/>][Error] Version [
  <sch:value-of select="document($pathToDocument)//cve:CVE//@specVersion"/>] of
  <sch:value-of select="$SpecToCheck"/>found; Version [
  <sch:value-of select="$MinVersion"/>] or later is required. The latest version of
  <sch:value-of select="$SpecToCheck"/>is not being used in the validation infrastructure. Regardless of the version indicated on the instance document, the validation infrastructure needs
to use a version of
  <sch:value-of select="$SpecToCheck"/>that is version [
  <sch:value-of select="$MinVersion"/>] or later. NOTE: This is not an error of the instance document but of the validation environment itself. The incorrect value was found in
  <sch:value-of select="document-uri(document($pathToDocument))"/>
    </sch:assert>
  </sch:rule>
</sch:pattern>
```

3.2 - ./Lib/ValidateValidationEnvSchema.sch

Code Description

This abstract pattern checks to see if the validation environment has at least the version / revision of the Schema as of the writing of this specification. The calling rule must pass in \$MinVersion, \$SpecToCheck, \$pathToDocument, \$RuleID.

Schematron Code

```
<!--
  This abstract pattern checks to see the version of a Schema is greater than or equal to a passed in parameter.

  $MinVersion      := the version that SpecToCheck must be equal to or greater than.
  $SpecToCheck     := Name the spec whose version in the infrastructure is being checked.
  $pathToDocument  := Relative path to the document xsd that has ther version string
  $RuleID          := The number of the rule in the concrete file.
-->

<sch:pattern xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
              abstract="true"
              id="ValidateValidationEnvSchema">
  <sch:rule id="ValidateValidationEnvSchema-R1" context="/">
    <sch:assert test="document($pathToDocument)//xsd:schema/@version castable as xs:double and document($pathToDocument)//xsd:schema/@version >= $MinVersion"
               flag="error"
               role="error">[
  <sch:value-of select="$RuleID"/>][Error] Version [
  <sch:value-of select="document($pathToDocument)//xsd:schema/@version"/>] of
  <sch:value-of select="$SpecToCheck"/>found; Version [
  <sch:value-of select="$MinVersion"/>] or later is required. The latest version of
  <sch:value-of select="$SpecToCheck"/>is not being used in the validation infrastructure. Regardless of the version indicated on the instance document, the validation infrastructure needs
to use a version of
  <sch:value-of select="$SpecToCheck"/>that is version [
  <sch:value-of select="$MinVersion"/>] or later. NOTE: This is not an error of the instance document but of the validation environment itself. The incorrect value was found in
  <sch:value-of select="document-uri(document($pathToDocument))"/>
    </sch:assert>
  </sch:rule>
</sch:pattern>
```

Chapter 4 - Schematron Schema

The top level Schematron file for SRC is in this section. This file imports all of the others and also defines many global variables they are all dependent on.

4.1 - ./SRC_XML.sch

Schematron Code

```
<!--UNCLASSIFIED-->
<?ICEA master?>
<!-- Notices - Distribution Notice:
      This document has been approved for Public Release and is available for use without restriction.
-->

<sch:schema queryBinding="xslt2"><!-- ***** -->
<!-- * Namespace declarations * -->
<!-- ***** -->

<sch:ns uri="urn:us:gov:ic:src" prefix="src"/>
      <sch:ns uri="urn:us:gov:ic:id" prefix="icid"/>
      <sch:ns uri="urn:us:gov:ic:tdf" prefix="tdf"/>
      <sch:ns uri="urn:us:gov:ic:intdis" prefix="intdis"/>
      <sch:ns uri="urn:us:gov:ic:ism" prefix="ism"/>
      <sch:ns uri="urn:us:gov:ic:cve" prefix="cve"/>
      <sch:ns uri="urn:us:gov:ic:edh" prefix="edh"/>
      <sch:ns uri="urn:us:gov:ic:virt" prefix="virt"/>
      <!-- ***** -->

<!-- * Abstract Rule and Pattern Includes * -->
<!-- ***** -->

<sch:include href="./Lib/ValidateValidationEnvSchema.sch"/>
      <sch:include href="./Lib/ValidateValidationEnvCVE.sch"/>
      <!-- ***** -->

<!-- * General Global Variables * -->
<!-- ***** -->

<sch:let name="timeZoneRegEx" value="'Z|[\+-]\d{2}:\d{2}'"/>
      <sch:let name="endsWithTimeZoneRegEx" value="concat('^.*',$timeZoneRegEx,'$')"/>
      <!--*****-->

<!-- (U) SRC ID Rules -->
<!--*****-->
<!--(U) TDFConstraints-->

<sch:include href="./Rules/TDFConstraints/SRC_ID_00002.sch"/>
      <sch:include href="./Rules/TDFConstraints/SRC_ID_00003.sch"/>
      <sch:include href="./Rules/TDFConstraints/SRC_ID_00004.sch"/>
      <!--(U) datesAndTimes-->

<sch:include href="./Rules/datesAndTimes/SRC_ID_00007.sch"/>
      <!--(U) elementBasedConstraints-->

<sch:include href="./Rules/elementBasedConstraints/SRC_ID_00010.sch"/>
      <!--(U) generalConstraints-->

<sch:include href="./Rules/generalConstraints/SRC_ID_00001.sch"/>
      <sch:include href="./Rules/generalConstraints/SRC_ID_00011.sch"/>
```



```

    <sch:include href="./Rules/generalConstraints/SRC_ID_00012.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00013.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00014.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00015.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00016.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00017.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00018.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00019.sch"/>
    <sch:include href="./Rules/generalConstraints/SRC_ID_00020.sch"/>
    <!--(U) nonNullConstraints-->

<sch:include href="./Rules/nonNullConstraints/SRC_ID_00009.sch"/>
    <!--(U) timeZoneIndicators-->

<sch:include href="./Rules/timeZoneIndicators/SRC_ID_00006.sch"/>
    <!--*****-->
<!-- (U) SRC Phases -->
<!--*****-->
</sch:schema>

    <!--UNCLASSIFIED-->
```

Chapter 5 - Removed Rules

All of the numbered Rules for SRC that have been removed are listed in this section. This section is just a reference for what rule numbers have been dropped. In many but not all cases there will be a reason listed. In all cases the version that the rule was dropped in is listed.

5.1 - `./Rules/deleted/SRC_ID_00005.sch`

Rule Description

[SRC-ID-00005][Error] Rule removed in V2015-AUG. Human Readable: Rule removed in V2015-AUG.

5.2 - `./Rules/deleted/SRC_ID_00008.sch`

Rule Description

[SRC-ID-00008][Error] Rule removed in V2015-AUG. Human Readable: Rule removed in V2015-AUG.