10TH EXPERT GROUP MEETING ON

Statistical Data and Metadata eXchange

JANUARY 25-28, 2021

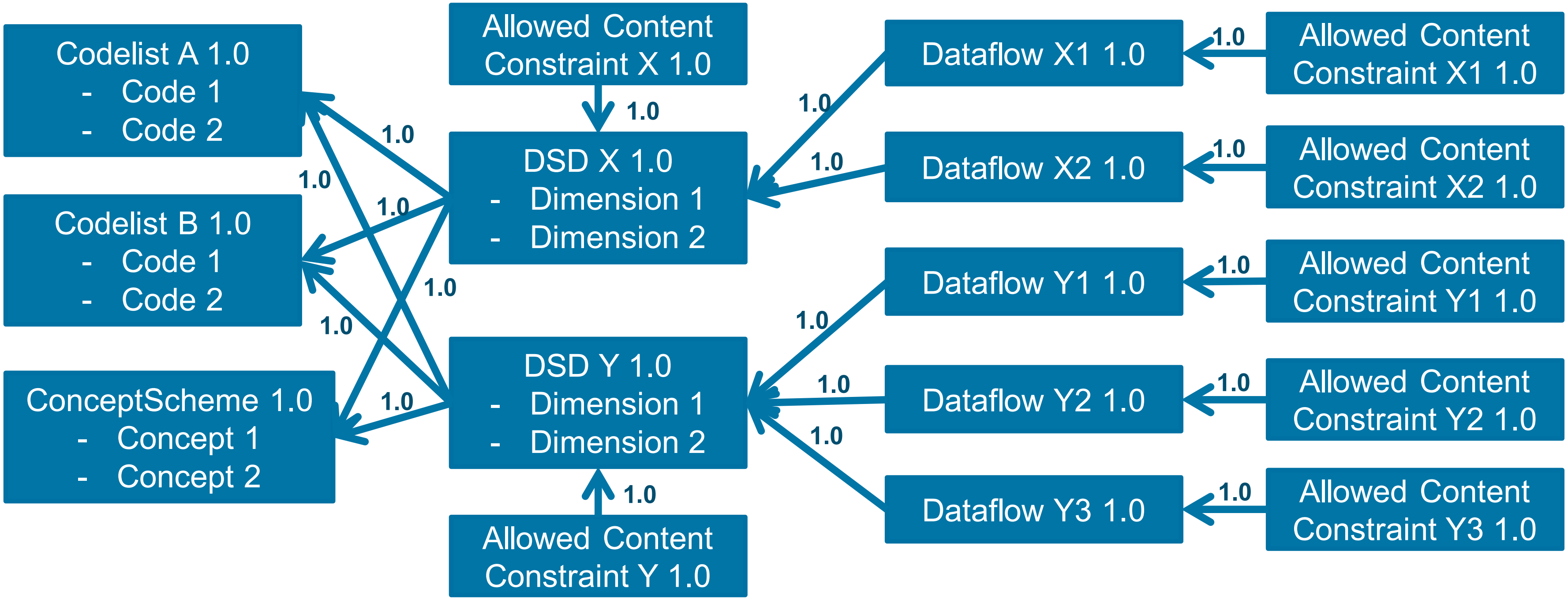
Semantic Versioning

JANUARY 25, 2021

Jens Dossé
Head of Smart Data Solutions Section/OECD
Co-Chair SDMX-TWG, Lead TF3

sdmx
Statistical Data and Metadata eXchange

Current versioning since SDMX 1.0



Codelist A 1.0
- Code 1
- Code 2

Codelist B 1.0
- Code 1
- Code 2

ConceptScheme 1.0
- Concept 1
- Concept 2

Allowed Content Constraint X 1.0

DSD X 1.0
- Dimension 1
- Dimension 2

DSD Y 1.0
- Dimension 1
- Dimension 2

Allowed Content Constraint Y 1.0

Dataflow X1 1.0

Dataflow X2 1.0

Dataflow Y1 1.0

Dataflow Y2 1.0

Dataflow Y3 1.0

Allowed Content Constraint X1 1.0

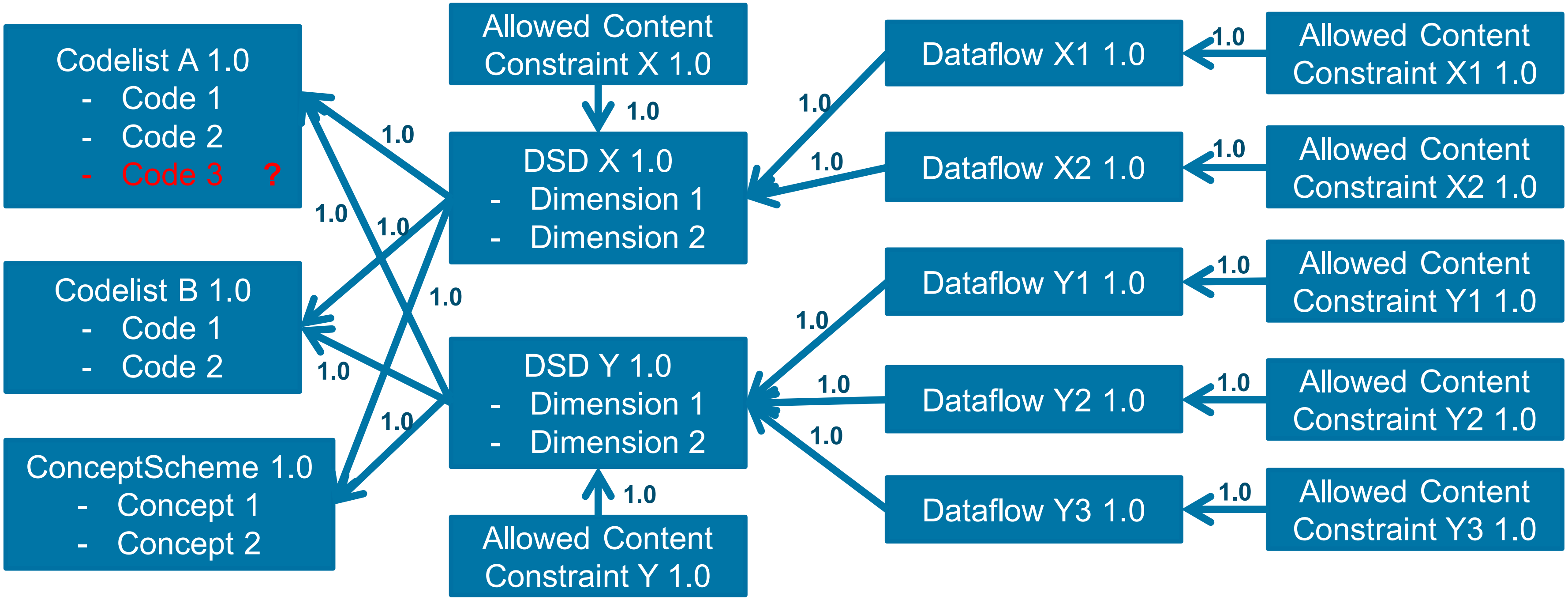
Allowed Content Constraint X2 1.0

Allowed Content Constraint Y1 1.0

Allowed Content Constraint Y2 1.0

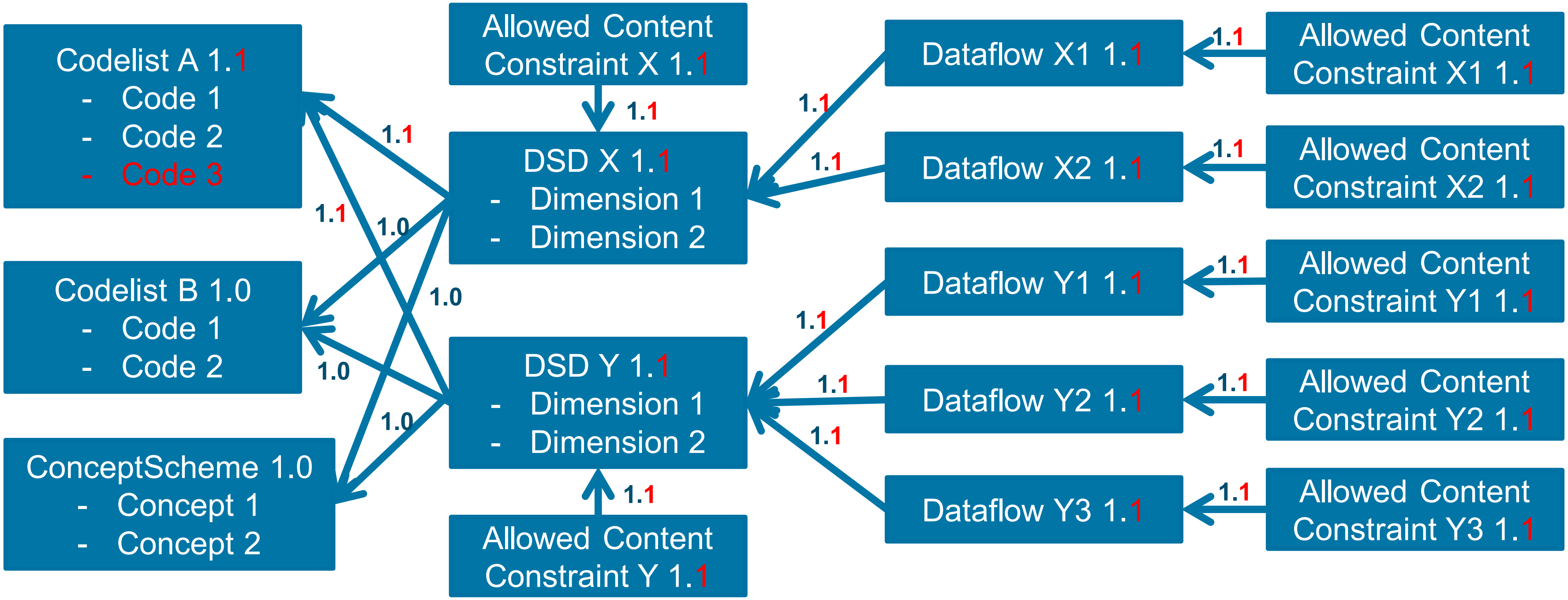
Allowed Content Constraint Y3 1.0

Current versioning since SDMX 1.0



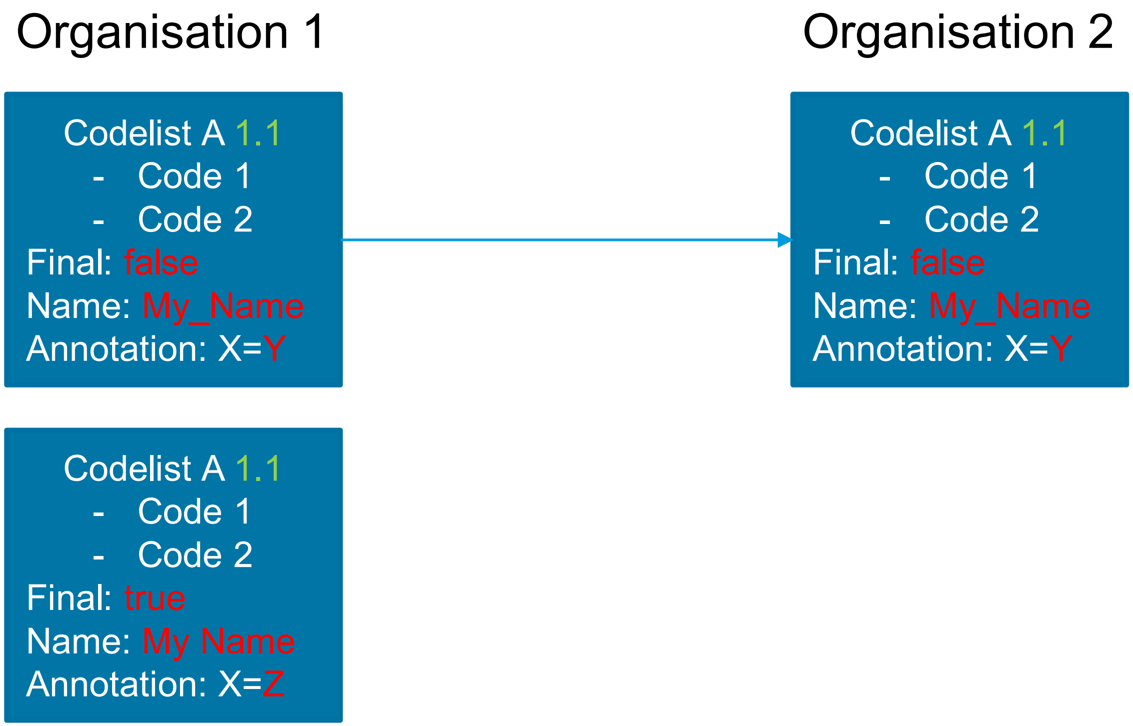
Statistical Data and Metadata eXchange

Current versioning since SDMX 1.0



Statistical Data and Metadata eXchange

Current versioning since SDMX 1.0



Organisation 1

Codelist A 1.1
- Code 1
- Code 2

Final: false
Name: My_Name
Annotation: X=Y

Codelist A 1.1
- Code 1
- Code 2

Final: true
Name: My Name
Annotation: X=Z

Organisation 2

Codelist A 1.1
- Code 1
- Code 2

Final: false
Name: My_Name
Annotation: X=Y

Versioning needs

- Make versioning optional
- When a use case needs versioning then make those versions fully reliable: use version number increases to communicate any change as well as the type of change (backward- and forward-compatibility)
- Reduce the burden of updates in referencing artefacts due to changes in referenced artefacts
- Allow a certain level of backward-compatibility with previous implementations

Semantic Versioning[1] in brief

- Versioning is optional; non-versioned artefacts have no version number and are allowed changing at any time in any way
- OrganisationsScheme artefacts will not use version numbers anymore
- The ‘isFinal’ property is removed

- Previous version numbers, if composed of two parts (MAJOR.MINOR) can still be used, without the ‘isFinal’ property
- These artefacts can also change at any time and in any way without a version number change
- Current non-standard governance processes continue to assure appropriate communication on changes in artefacts

- Truly versioned artefacts must use semantic versioning based on 3 or 4 version parts: MAJOR.MINOR.PATCH[-EXTENSION]
- Semantically versioned artefacts are immutable
- For such artefacts increase:
  1. MAJOR version when backwards incompatible artefact changes are made,
  2. MINOR version when artefact elements are added in a backwards compatible manner, or
  3. PATCH version when only backwards compatible artefact property changes are made
- When incrementing a version part, the right-hand side parts are 0-ed. Extensions can be added, changed or dropped
- When incrementing a version, changes within the version change scope are permitted until the new version is released
- MAJOR.MINOR.PATCH >= 0.0.0
- 0.MINOR.PATCH is for initial modelling and anything may change at any time. Version 1.0.0 defines the first stable artefact
- A version EXTENSION is for intermediate modelling and changes and can change at any time within the allowed scope of the version increment from the previous version. The recommended standard extension is “-draft”.
- Precedence is determined by the first difference when comparing each of these identifiers from left to right. When major, minor, and patch are equal, an extended version has lower precedence than a stable version.
- The reasons for version changes MAY be documented in brief form in an artefact's annotation of type "CHANGELOG".

sdmx
Statistical Data and Metadata eXchange

[1] semver.org

Dependency Mgmt. in brief

1. Wildcarding MAJOR, MINOR or PATCH version parts in SDMX 3.0 artefact references using “+” as suffix:
   - X+.Y.Z references the currently latest available version >= X.Y.Z

     Use case: an SDMX Categorisation with a dataflow “2+.3.1” reference: references the currently latest available version >= “2.3.1”
   - X.Y+.Z references the currently latest available backwards compatible version >= X.Y.Z

     Use case: an SDMX DSD with a codelist “2.3+.1” reference: references the currently latest available version >= “2.3.1” and < "3.0.0" (all backwards compatible versions >= "2.3.1")
   - X.Y.Z+ references the currently latest available forwards and backwards compatible version >= X.Y.Z

     Use case: an SDMX Allowed ContentConstraint referencing a dataflow “2.3.1+”:
     references the currently latest available version >= “2.3.1” and < "2.4.0" (all forwards and backwards compatible versions >= "2.3.1")

2. Non-versioned and 2-digit-version artefacts can reference any artefacts

3. Semantically versioned artefacts must only reference other semantically versioned artefacts.
   - Wildcarded references in a stable artefact implicitly target only future stable versions of the referenced artefacts within the defined wildcard scope.

     Example: "AGENCY_ID:CODELIST_ID(2.3+.1)" in an artefact "AGENCY_ID:DSD_ID(2.2.1)” resolves to artefact "AGENCY_ID:CODELIST_ID(2.4.3)" if that was currently the latest available stable version.
   - Wildcarded references in a version-extended artefact implicitly target future stable and version-extended versions of the referenced artefacts within the defined wildcard scope.

     Example: The reference to "AGENCY_ID:CODELIST_ID(2.3+.1)" in an artefact "AGENCY_ID:DSD_ID(2.2.1-draft)” resolves to artefact "AGENCY_ID:CODELIST_ID(2.5.0-draft)" if that was currently the latest available version.
   - References to specific version-extended artefacts MAY be used, but those cannot be combined with a wildcard.

     Example: The reference to "AGENCY_ID:CODELIST_ID(2.5.0-draft)" in an artefact "AGENCY_ID:DSD_ID(2.2.1)” resolves to artefact "AGENCY_ID:CODELIST_ID(2.5.0-draft)", which might be subject to continued backwards compatible changes.

REST API changes

The special keywords **latest** and **all** are replaced by:

Feature	Character
All	*
Latest stable	+
Latest draft	~
OR	,
Subpart separator	.
Multivalues attribute	+

Assuming the availability of an artefact with the following versions: 1.0, 1.0.0, 1.1.0, 1.2.0, 1.2.1, 1.2.2, 1.3.0, 1.3.1-draft, 2.0.0, 2.1.0-draft

All versions

*	This returns all versions
1.*	This returns 1.0
1.*.0	This returns all versions, except 1.0, 2.0.0, 2.1.0-draft
1.3.*	This returns 1.3.0 and 1.3.1-draft

Latest stable version

+	This returns version 2.0.0
1.3.+	This returns version 1.3.0

Latest possibly unstable version

~	This returns version 2.1.0-draft
1.~	This returns version 1.0
1.~.0	This returns version 1.3.1-draft
1.1.~	This returns version 1.1.0

All versions above and including a minimum version

1.2*.0	This returns 1.2.0, 1.2.1, 1.2.2, 1.3.0 and 1.3.1-draft

Latest stable version above a minimum version

1+.2.0	This returns 2.0.0
1.2+.0	This returns 1.3.0
1.2.0+	This returns 1.2.2

Latest possibly unstable version above a minimum version

1~.2.0	This returns 2.1.0-draft
1.2~.0	This returns 1.3.1-draft
1.2.0~	This returns 1.2.2

OR operator

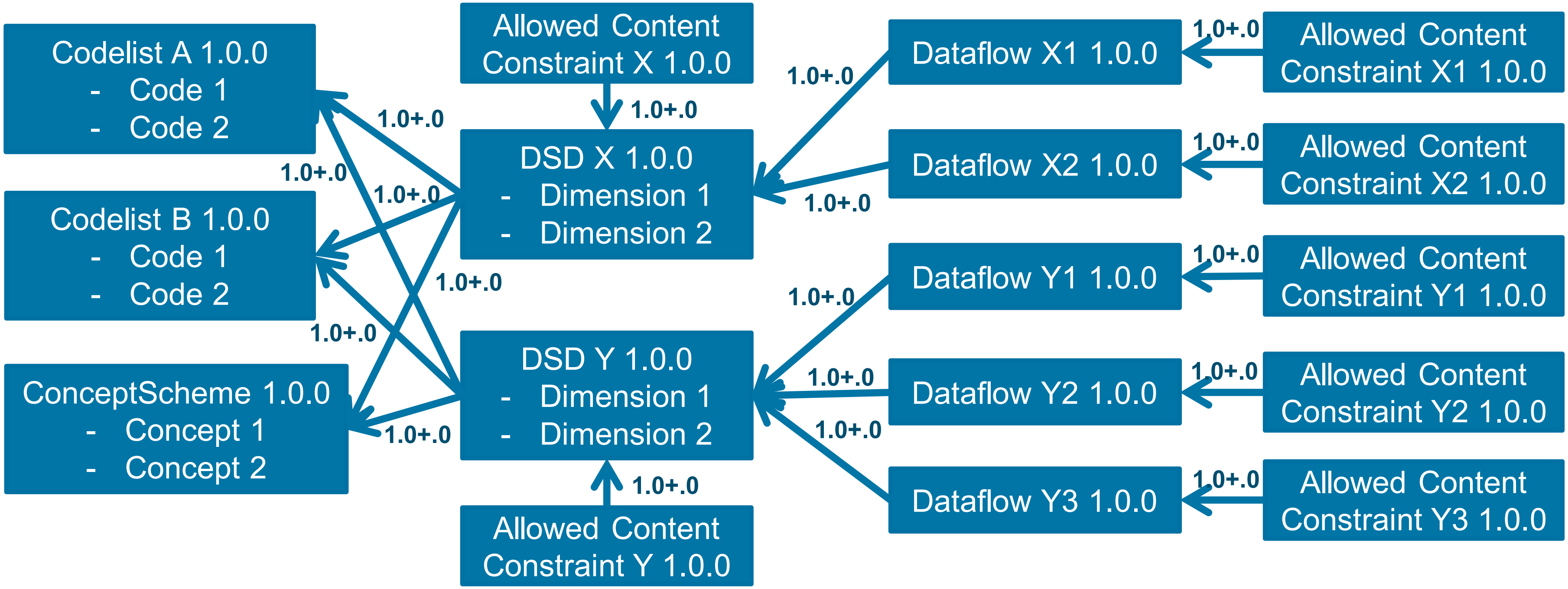
1~.2.0,1.2.0+	This returns 2.1.0-draft and 1.2.2
+,1.2.1*	This returns 2.0.0, 1.2.1, and 1.2.2

Specific versions

1.0	This returns version 1.0
1.0.0	This returns version 1.0.0
1.3.1-draft	This returns version 1.3.1-draft

sdmx
Statistical Data and Metadata eXchange

Semantic Versioning in SDMX 3.0(.0)



Codelist A 1.0.0
- Code 1
- Code 2

Codelist B 1.0.0
- Code 1
- Code 2

ConceptScheme 1.0.0
- Concept 1
- Concept 2

Allowed Content Constraint X 1.0.0

DSD X 1.0.0
- Dimension 1
- Dimension 2

DSD Y 1.0.0
- Dimension 1
- Dimension 2

Allowed Content Constraint Y 1.0.0

Dataflow X1 1.0.0

Dataflow X2 1.0.0

Dataflow Y1 1.0.0

Dataflow Y2 1.0.0

Dataflow Y3 1.0.0

Allowed Content Constraint X1 1.0.0

Allowed Content Constraint X2 1.0.0

Allowed Content Constraint Y1 1.0.0

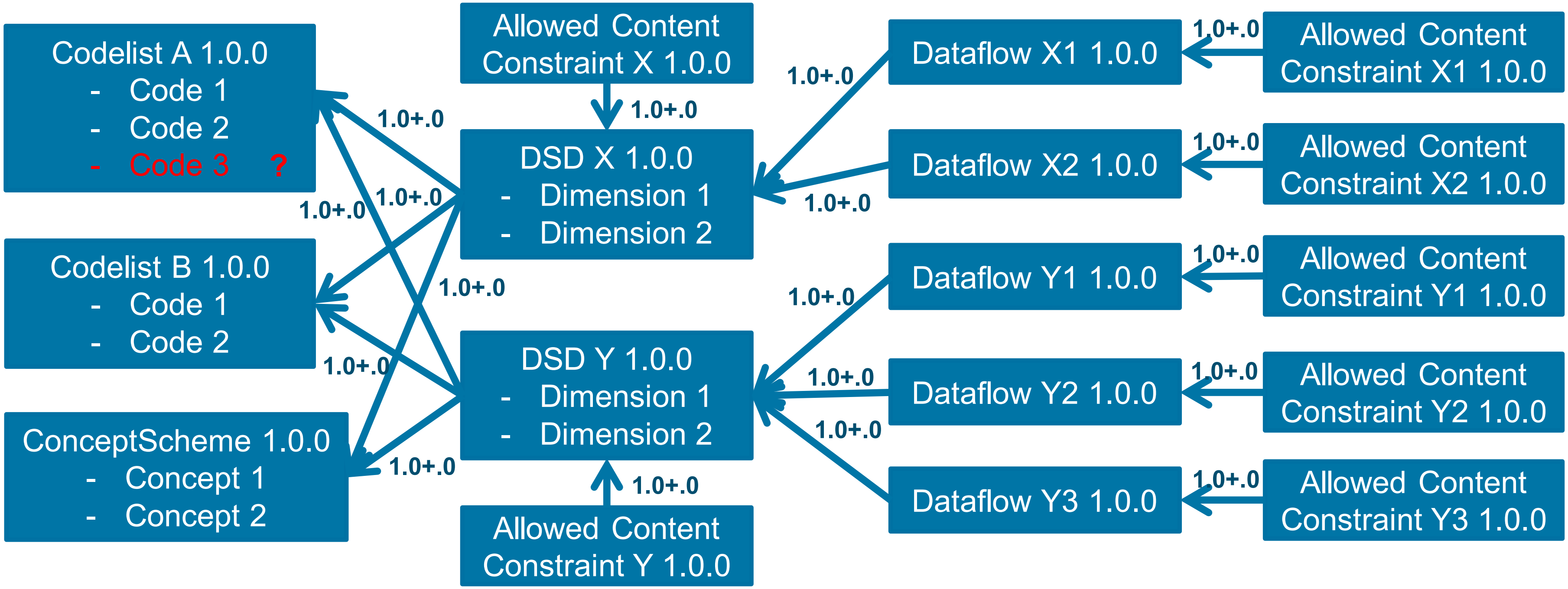
Allowed Content Constraint Y2 1.0.0

Allowed Content Constraint Y3 1.0.0

sdmx

Statistical Data and Metadata eXchange

Semantic Versioning in SDMX 3.0(.0)



Codelist A 1.0.0
- Code 1
- Code 2
- Code 3 ?

Codelist B 1.0.0
- Code 1
- Code 2

ConceptScheme 1.0.0
- Concept 1
- Concept 2

Allowed Content Constraint X 1.0.0

DSD X 1.0.0
- Dimension 1
- Dimension 2

DSD Y 1.0.0
- Dimension 1
- Dimension 2

Allowed Content Constraint Y 1.0.0

Dataflow X1 1.0.0

Dataflow X2 1.0.0

Dataflow Y1 1.0.0

Dataflow Y2 1.0.0

Dataflow Y3 1.0.0

Allowed Content Constraint X1 1.0.0

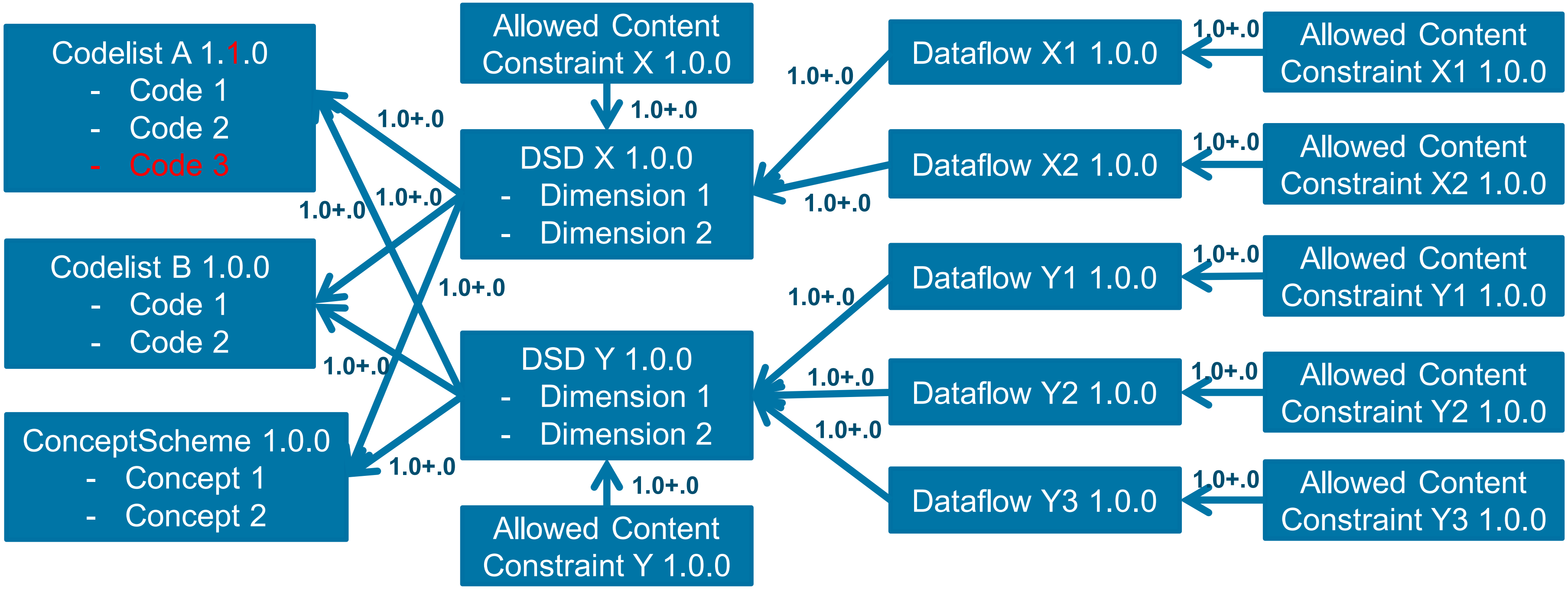
Allowed Content Constraint X2 1.0.0

Allowed Content Constraint Y1 1.0.0

Allowed Content Constraint Y2 1.0.0

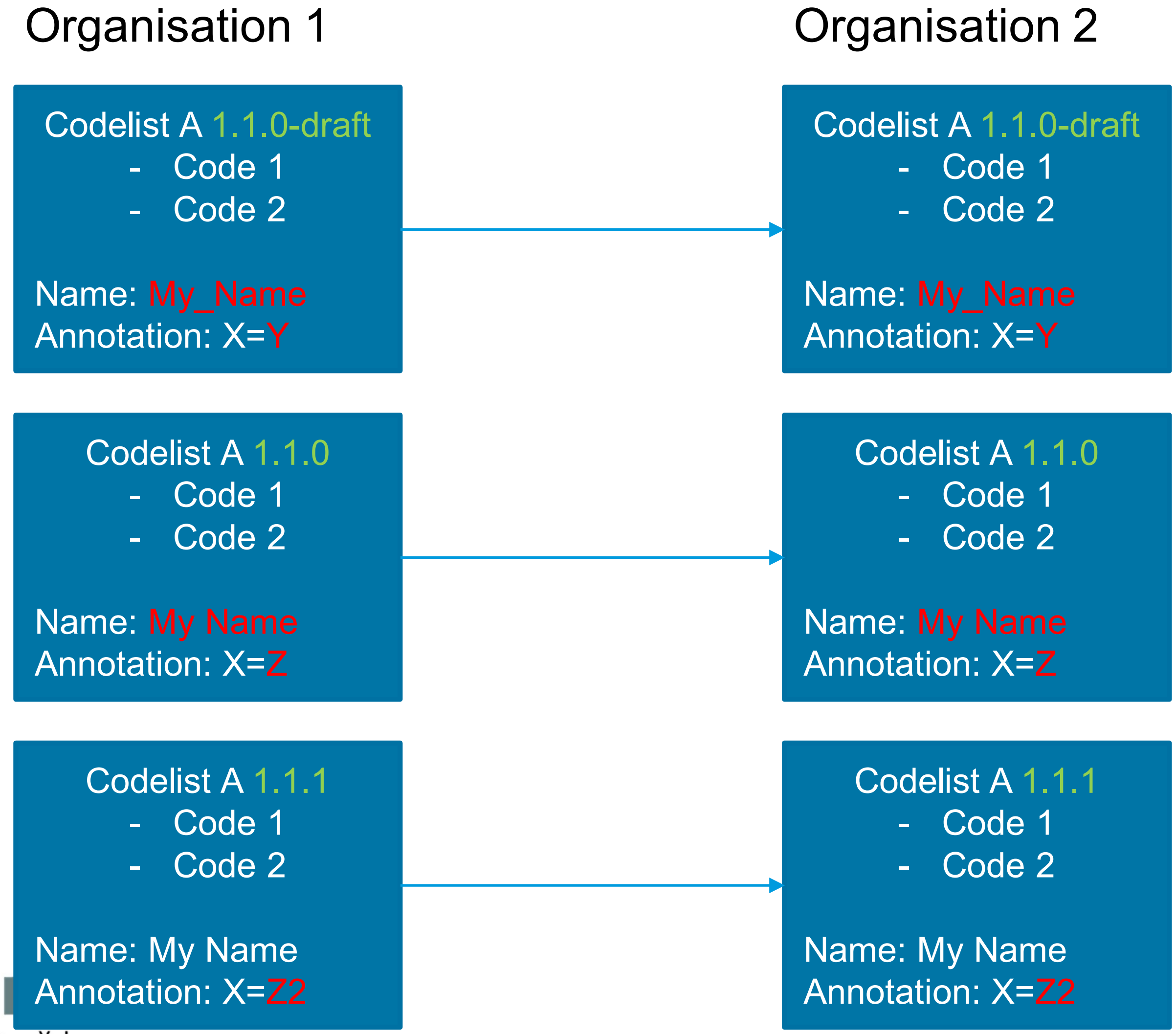
Allowed Content Constraint Y3 1.0.0

Semantic Versioning in SDMX 3.0(.0)



sdmx
Statistical Data and Metadata eXchange

Semantic Versioning in SDMX 3.0(.0)



Organisation 1

Organisation 2

Codelist A 1.1.0-draft
- Code 1
- Code 2

Name: My_Name
Annotation: X=Y

Codelist A 1.1.0-draft
- Code 1
- Code 2

Name: My_Name
Annotation: X=Y

Codelist A 1.1.0
- Code 1
- Code 2

Name: My Name
Annotation: X=Z

Codelist A 1.1.0
- Code 1
- Code 2

Name: My Name
Annotation: X=Z

Codelist A 1.1.1
- Code 1
- Code 2

Name: My Name
Annotation: X=Z2

Codelist A 1.1.1
- Code 1
- Code 2

Name: My Name
Annotation: X=Z2

Thank you.

Statistical Data and Metadata eXchange