FUSION REGISTRY

FUSION REGISTRY
VERSION 9

Item Validity

This guide describes how the Registry supports Item Validity allowing for codes and concepts to be valid for specific periods of time.

Fusion Registry: 9.5.x
Document Last Updated: August 10th 2018

Version 20180810

1 Contents

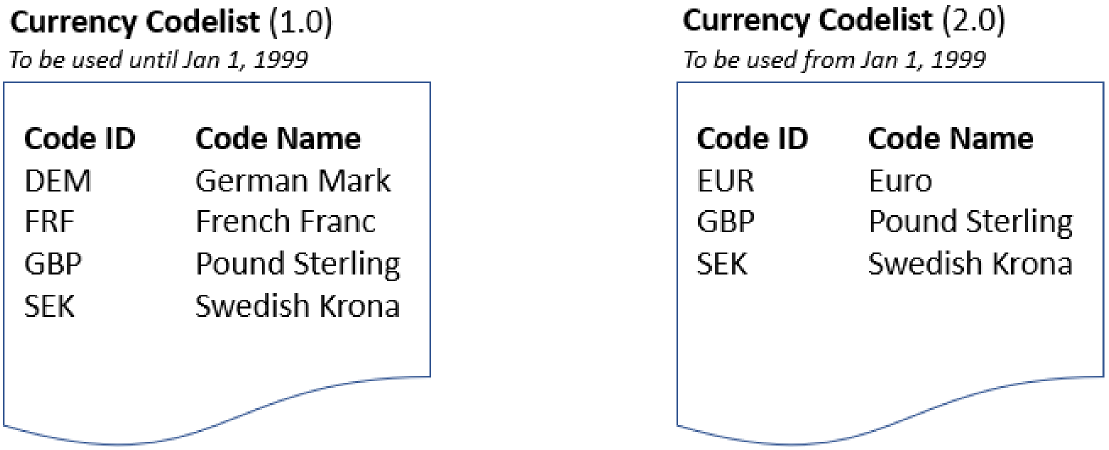
2 Overview

Item Validity allows for codes and concepts to be specified as being valid for specific periods of time. This is useful for when codes in a Codelist are need to be re-assigned, or discontinued. Without Item Validity the way to achieve this would be to create a new Codelist which has the changes and then migrate the users of the existing Codelist to use the new Codelist.

Item Validity can be applied to Concepts in a Concept Scheme or Codes in a Codelist, a Hierarchical Codelist or a Codelist Map.

2.1 Discontinuing a Code

If a Codelist contains codes that are no longer required then without using Item Validity, a new Codelist must be created which only contains the valid codes. For example, consider the change to European Currencies with the introduction of the Euro.



Currency Codelist (1.0)
To be used until Jan 1, 1999

Code ID	Code Name
DEM	German Mark
FRF	French Franc
GBP	Pound Sterling
SEK	Swedish Krona

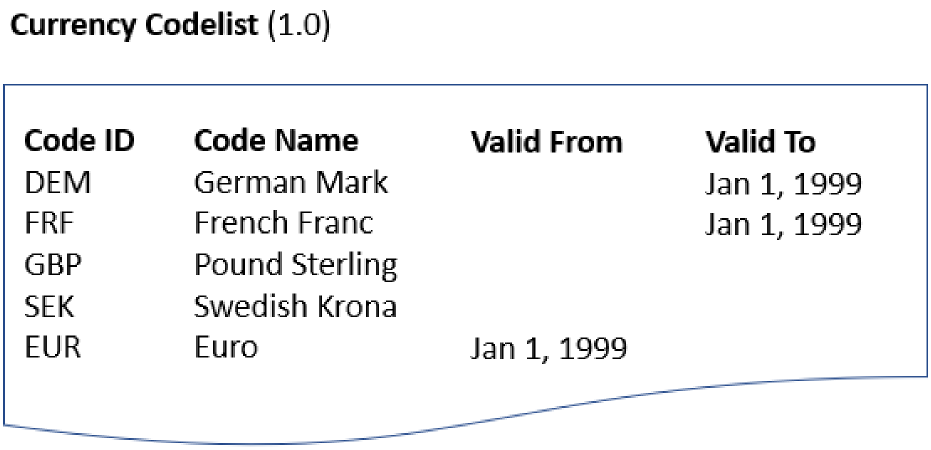
Currency Codelist (2.0)
To be used from Jan 1, 1999

Code ID	Code Name
EUR	Euro
GBP	Pound Sterling
SEK	Swedish Krona

Figure 1- Two Currency Codelists required to show migration to the Euro

The Currency Codelist (version 1.0) contains the codes for all currencies. With the introduction of the Euro and the discontinuation of existing currencies (DEM and FRF), a new Codelist must be created (version 2.0) which contains only the codes that are to be used. Each version of the Codelist should have a date that states when it should be used. For Codelist version 1.0, this is up to Jan 1, 1999 and for Codelist version 2.0 this is from Jan 1, 1999. Any codes that don't change are reflected in both Codelists.

Item Validity can overcome the overhead of creating a new Codelist. In the existing Codelist, items can be discontinued by adding a date stating when it is Valid To. Since DEM and FRF need to be discontinued, these are given a "Valid To" date.

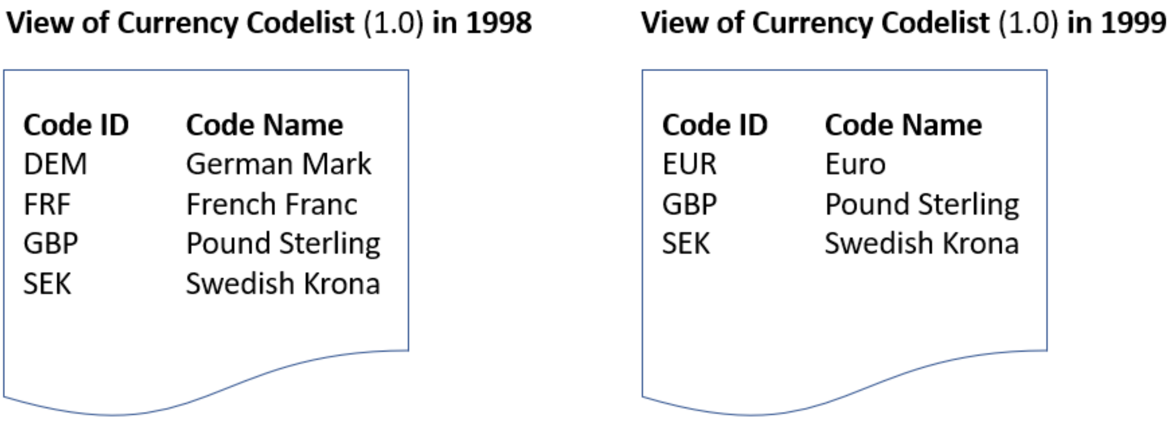


Currency Codelist (1.0)

Code ID	Code Name	Valid From	Valid To
DEM	German Mark		Jan 1, 1999
FRF	French Franc		Jan 1, 1999
GBP	Pound Sterling		
SEK	Swedish Krona		
EUR	Euro	Jan 1, 1999	

Figure 2- A single Currency Codelist with Item Validity

For the new Euro code, this can be added to the Codelist with a date specifying when this code is Valid From. So now the Codelist will have different codes for different periods of time.



View of Currency Codelist (1.0) **in 1998**

Code ID	Code Name
DEM	German Mark
FRF	French Franc
GBP	Pound Sterling
SEK	Swedish Krona

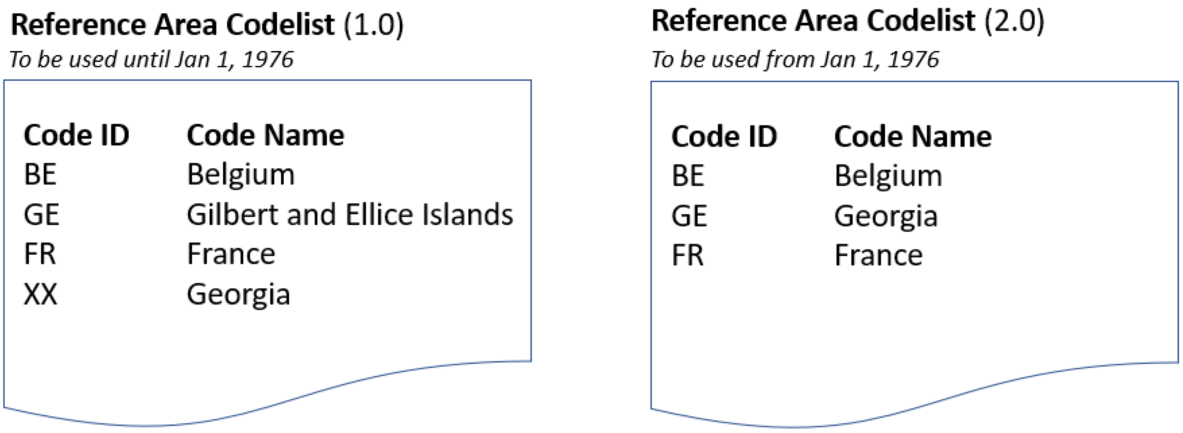
View of Currency Codelist (1.0) **in 1999**

Code ID	Code Name
EUR	Euro
GBP	Pound Sterling
SEK	Swedish Krona

Figure 3 - The Item Validity Currency Codelist viewed at two separate points in time

2.2 Reassigning a Code

If a code needs to be re-assigned because its use has changed, without the use of Item Validity a new Codelist must be created. Consider the example of the Gilbert and Ellice Islands which was a territory that existed until 1976. The following illustrates an example Codelist of Reference Area:



Reference Area Codelist (1.0)

To be used until Jan 1, 1976

Code ID	Code Name
BE	Belgium
GE	Gilbert and Ellice Islands
FR	France
XX	Georgia

Reference Area Codelist (2.0)

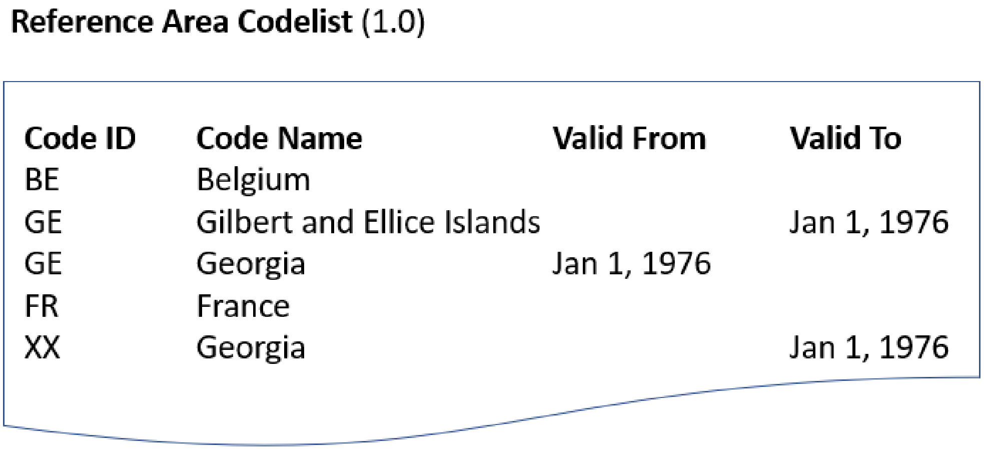
To be used from Jan 1, 1976

Code ID	Code Name
BE	Belgium
GE	Georgia
FR	France

Figure 4 - Two Reference Area Codelists required to show territories

With the dissolution of the Gilbert and Ellice Islands, the code "GE" is re-assigned to Georgia and Georgia's existing code ("XX") is discontinued.

With the use of Item Validity a single Codelist can discontinue the use of the code "GE" to mean the Gilbert and Ellice Islands and re-assign it to Georgia. The use of Valid From and Valid To dates means that codes can be terminated and re-assigned as desired.



Reference Area Codelist (1.0)

Code ID	Code Name	Valid From	Valid To
BE	Belgium		
GE	Gilbert and Ellice Islands		Jan 1, 1976
GE	Georgia	Jan 1, 1976	
FR	France		
XX	Georgia		Jan 1, 1976

Figure 5 - A single Reference Area Codelist with Item Validity

It is important to note that in the example above the Reference Area Codelist has two entries for the code "GE". This is not permitted in a "standard" Codelist. However since each of the GE entries has a Validity Period and these Validity Periods do not overlap, this is permissible. At no point in time are both codes valid simultaneously. If a further re-assignment to the code "GE" it is possible to have more entries for "GE" provided that the Validity Periods do not overlap.

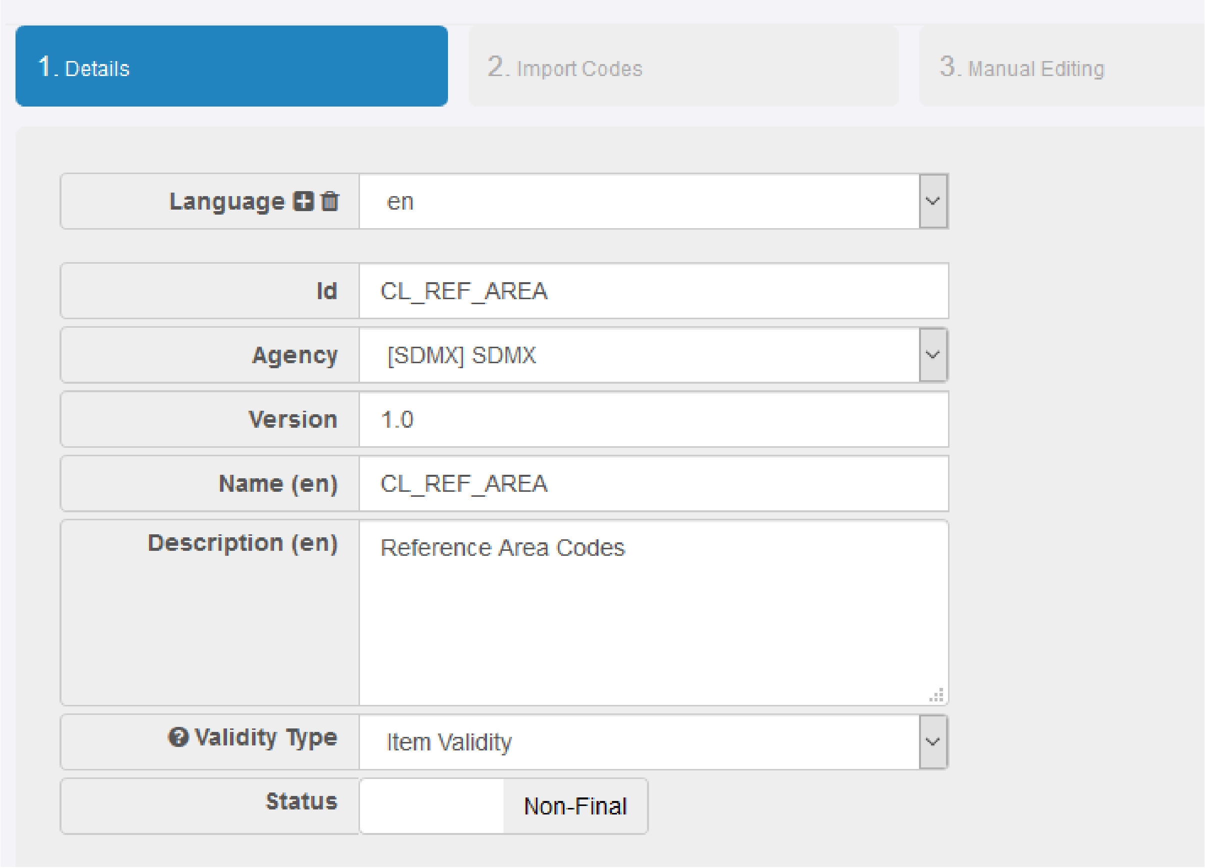
3 Assigning Item Validity

3.1 Codelist Wizard – Stating the Codelist Type

The Codelist Wizard in Fusion Registry allows the user to create and edit Codelists. In Step 1 of the wizard aside from the standard elements that make up a structure, the Codelist wizard adds a further choice control which allows a Codelist to be defined in one of three forms:

- A Standard Codelist – this allows a Validity Period to be specified on the Codelist which specifies the Codelist's validity to a single date range, although this may be open-ended. All other structural metadata in the Registry has this concept.
- An Item-Validity Codelist – this is where the codes in the Codelist may be assigned individual validity periods. This means that over time, individual codes in the Codelist may become valid or invalid as specified by their code definition. This is particularly useful when re-assigning codes for a different semantic meaning.
- A Codelist Template – this is used to generate other Codelists from a single template. Like an Item-Validity Codelist, the Codelist Template permits validity at the item level. Please refer to section 5 for an explanation of a Codelist Template.

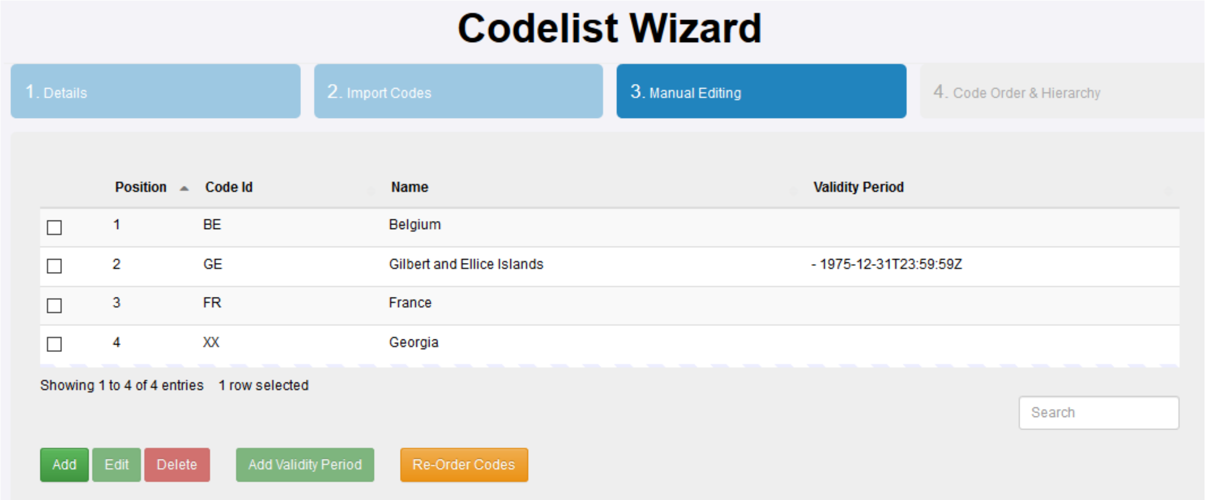
It is not possible to assign Validity Periods to both the Codelist and individual codes.



Language	en
Id	CL_REF_AREA
Agency	[SDMX] SDMX
Version	1.0
Name (en)	CL_REF_AREA
Description (en)	Reference Area Codes
Validity Type	Item Validity
Status	Non-Final

Figure 6- Step 1 of the Codelist Wizard - the user has chosen Item Validity

3.2 Codelist Wizard – Assigning a Validity Period to an Item



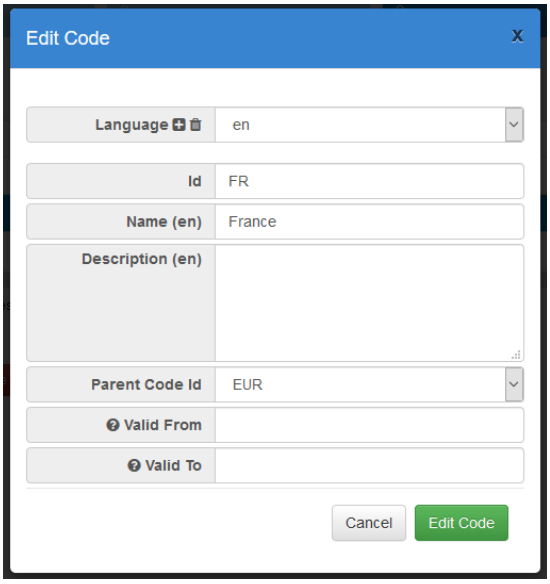
Codelist Wizard

	Position	Code Id	Name	Validity Period
□	1	BE	Belgium	
□	2	GE	Gilbert and Ellice Islands	- 1975-12-31T23:59:59Z
□	3	FR	France	
□	4	XX	Georgia	

Showing 1 to 4 of 4 entries 1 row selected

Figure 7 - Codelist Wizard Step 3

The third step of the wizard allows the user to manually add, edit, and delete Codes in the Codelist. For editing and deleting codes, if the Codelist is of type "Item Validity" then the edit window will have entries for "Valid From" and "Valid To". If it is a "standard" Codelist these fields will not be present.



Edit Code

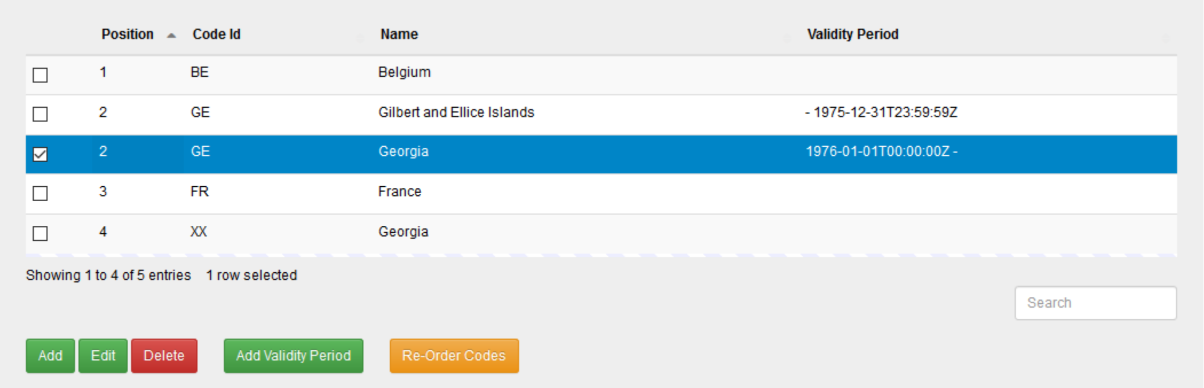
Language	en
Id	FR
Name (en)	France
Description (en)	
Parent Code Id	EUR
Valid From	
Valid To	

Figure 8 - Codelist Wizard Step 3 - Editing a Code in an "Item Validity" Codelist

Simply editing the "Valid From" and "Valid To" fields is enough to specify Item Validity to the currently edited item. The input takes SDMX Datetimes, so such values as 2001-Q1 and 2002-W50 are permitted, but if you prefer the

Note that unless a full date and time is specified, the Registry will infer that start dates always happen at the start of a period and that end dates will always be the end of a period. For example, typing 2001 into the "Valid From" field will result in a value of 2001-01-01T00:00:00 whereas entering 2001 into the "Valid To" field will result in a value of 2001-12-31T23:59:59. To avoid ambiguity, the Registry will always expand these values to a full date and time.

If a code has Item Validity, clicking the button “Add Validity Period” will bring up the edit window but allows for a second item with the same Code Id to be created, provided that the Validity Periods of the items with the same code, do not overlap.



	Position	Code Id	Name	Validity Period
□	1	BE	Belgium	
□	2	GE	Gilbert and Ellice Islands	- 1975-12-31T23:59:59Z
☑	2	GE	Georgia	1976-01-01T00:00:00Z -
□	3	FR	France	
□	4	XX	Georgia	

Showing 1 to 4 of 5 entries 1 row selected

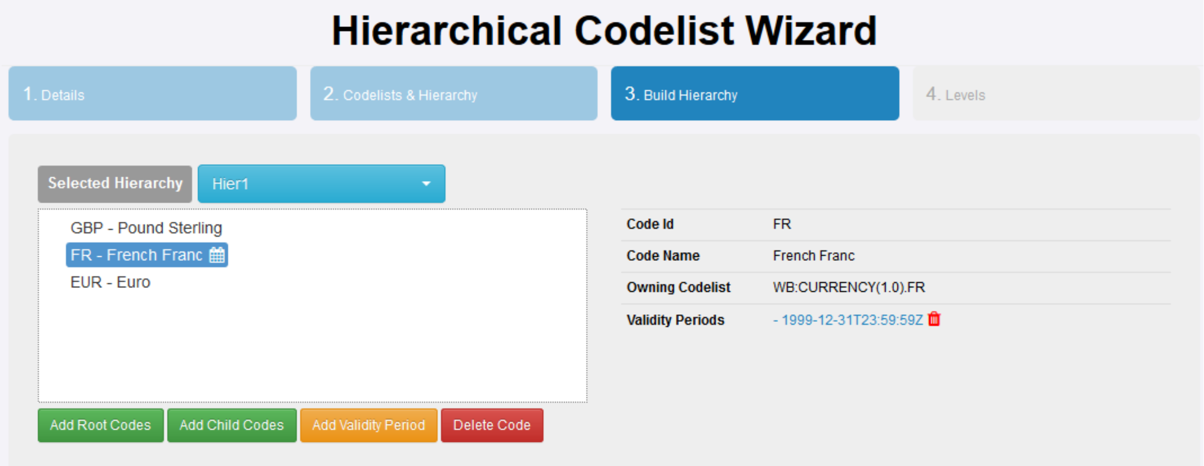
Figure 9- Duplicate Codes are permitted if the Validity Periods do not overlap

3.3 Concept Scheme Wizard

To add Validity Periods to items in a Concept Scheme, the Concept Scheme wizard follows the same approach as the Codelist Wizard, except that on Step 3, it is not permissible to assign multiple validity periods to the same code.

3.4 Hierarchical Codelist Wizard

The Hierarchical Codelist does not permit the setting of an item type in Step 1, like the Codelist and Concept Scheme wizard. Instead on step 3 Validity Periods are assigned to an individual item via the control “Add Validity Period”.



Hierarchical Codelist Wizard

Selected Hierarchy: Hier1

GBP - Pound Sterling
FR - French Franc
EUR - Euro

Code Id	FR
Code Name	French Franc
Owning Codelist	WB:CURRENCY(1.0).FR
Validity Periods	- 1999-12-31T23:59:59Z

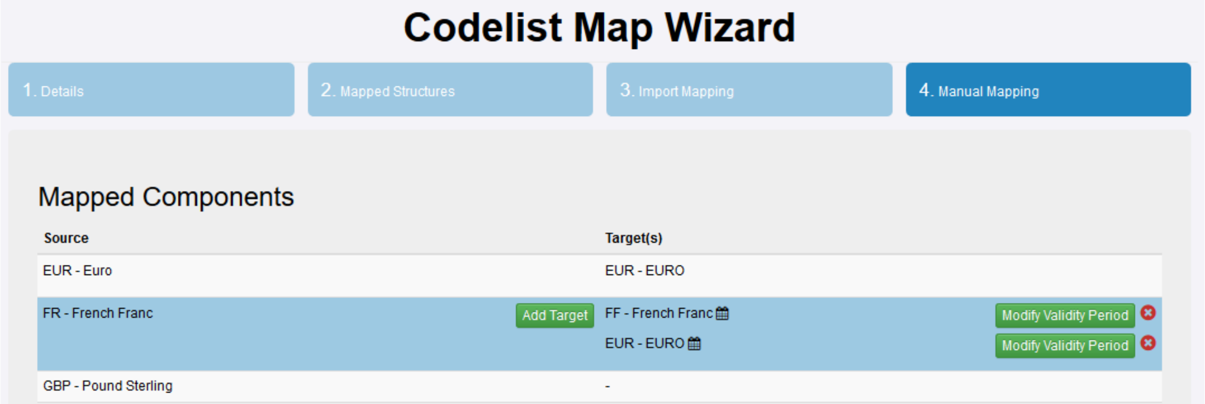
Figure 10

It is permissible to add multiple Validity Periods to a particular item if desired.

Also note that a child code cannot have a validity period “wider” than its parent. If a parent code of “EUR” exists which has a validity of the year 1999 onwards, any children of EUR can only have validity periods from 1999 onwards.

3.5 Codelist Map Wizard

Step 4 of the Codelist Map Wizard allows for the definition of a Code from a source to a target. On this step it is also possible to add a Validity Period to a single mapping

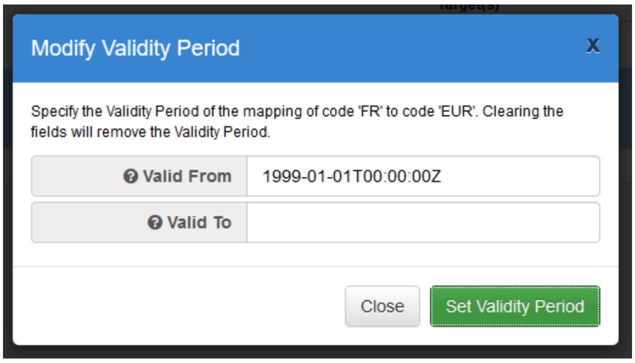


Mapped Components

Source	Target(s)
EUR - Euro	EUR - EURO
FR - French Franc	FF - French Franc EUR - EURO
GBP - Pound Sterling	-

Figure 11 – Codelist Map Wizard showing the mapping of FR to FF and FR to EUR for two different validity periods

If a mapping has a validity period, it is shown with a calendar icon to state that this mapping is date dependant. To assign, remove or edit a Validity Period for a mapping, activate the button labelled “Modify Validity Period” and the following modal is displayed:



Specify the Validity Period of the mapping of code 'FR' to code 'EUR'. Clearing the fields will remove the Validity Period.

Valid From: 1999-01-01T00:00:00Z

Valid To:

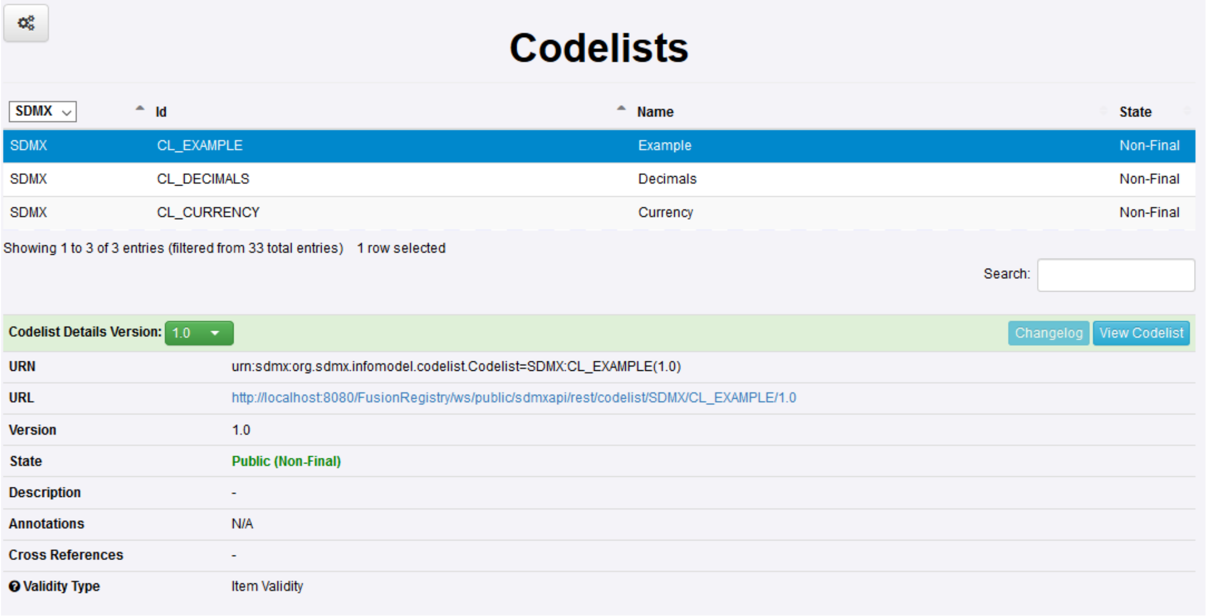
Figure 12 - showing the modification of the Validity Period for the mapping FR to EUR

The input for each is an SDMX datetime. Clearing the values and clicking “Set Validity Period” will remove the mapping.

4 Viewing Item Validity

4.1 Viewing a Codelist

When viewing the list of Codelists in the system, the Validity Type (as defined on Stage 1 of the Codelist Wizard) is displayed in the lower half of the screen. This value will either be “Standard”, “Item Validity”, “Template” or “Generated”.



Codelists

SDMX	Id	Name	State
SDMX	CL_EXAMPLE	Example	Non-Final
SDMX	CL_DECIMALS	Decimals	Non-Final
SDMX	CL_CURRENCY	Currency	Non-Final

Showing 1 to 3 of 3 entries (filtered from 33 total entries) 1 row selected

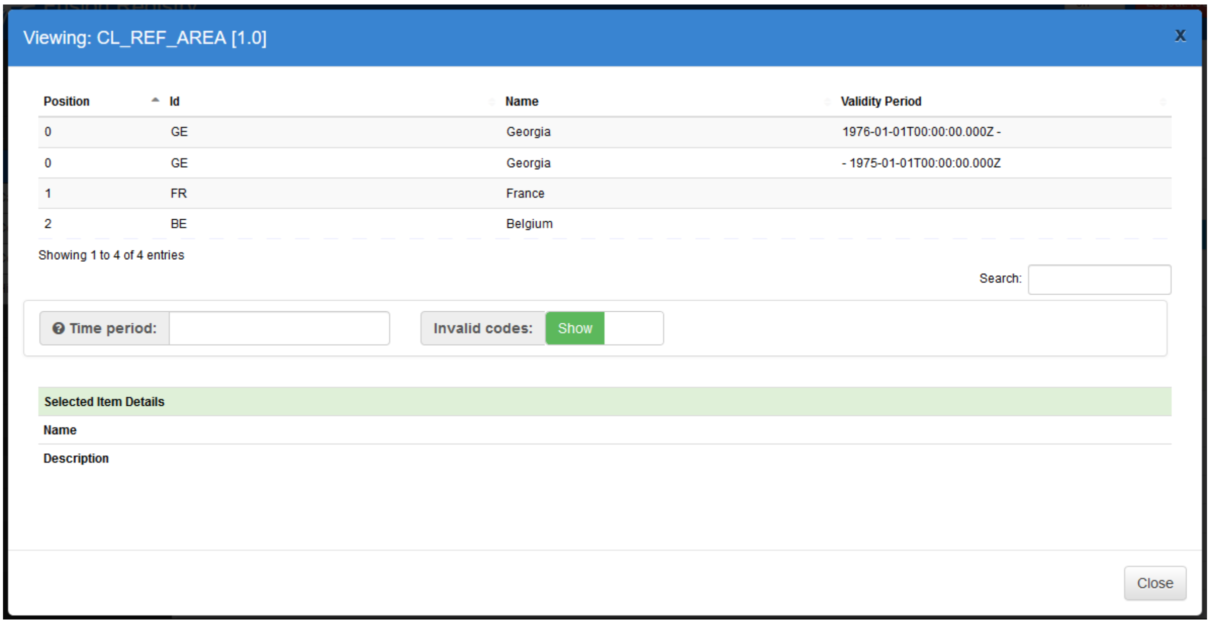
Search:

Codelist Details Version: 1.0

URN	urn:sdmx:org.sdmx.infomodel.codelist.Codelist=SDMX:CL_EXAMPLE(1.0)
URL	http://localhost:8080/FusionRegistry/ws/public/sdmxapi/rest/codelist/SDMX/CL_EXAMPLE/1.0
Version	1.0
State	Public (Non-Final)
Description	-
Annotations	N/A
Cross References	-
Validity Type	Item Validity

Figure 13- showing the Codelist "CL_EXAMPLE" which is of type "Item Validity"

When the user clicks “View Codelist”, if the Codelist does not have codes with validity at the item level, then the modal dialog shows a table of codes with the columns: Position, Id, Name and Description. If the Codelist has codes with validity at the item level then the Description column is not shown in the table and the Validity Period is shown instead.



Viewing: CL_REF_AREA [1.0]

Position	Id	Name	Validity Period
0	GE	Georgia	1976-01-01T00:00:00.000Z -
0	GE	Georgia	- 1975-01-01T00:00:00.000Z
1	FR	France	
2	BE	Belgium	

Showing 1 to 4 of 4 entries

Search:

Time period:

Invalid codes: Show

Selected Item Details

Name

Description

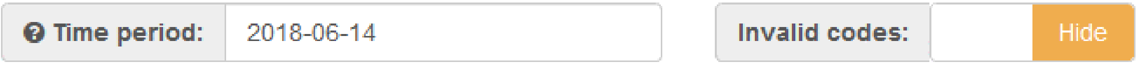
Figure 14 - Viewing a Codelist which has codes with Validity Periods

Note that there may be duplicate rows for the same ID and that for these rows, the position is the same. Any rows which are considered invalid for today will be rendered in grey text.

In the middle of the screen are controls for the time period and display of Invalid Codes.

4.1.1 View Time

The "View Time" is the time that the user is viewing this Codelist. By default, this will be blank. A blank value infers the current time.



Time period: 2018-06-14

Invalid codes: Hide

Figure 15 - Detail of the Time controls

The user may select a datetime by using the Time Period input field. This field allows the user to type in any valid SDMX Datetime Value (e.g. 2001-Q4) and when return is pressed, the table updates making any of the codes which are not valid for that time turn grey. This allows the user to see the state of a Codelist for any point in time.

There is a slide control for the hiding or showing of "invalid codes" (i.e. codes which are not valid for the current time period). By default, this is set to show invalid codes – rendered in grey. If this is set to "hide" then only currently valid codes are shown (rows which would have been rendered in grey are now not shown).

For viewing Hierarchical Codelists, Code Maps and Concept Schemes the same controls as in the Codelist will be available for those structures that have items with Validity Periods.

5 Codelist Templates

A Codelist Template allows Codelists to be generated from a definition, and for that definition to persist in the Fusion Registry to allow the Codelists to be re-generated at a future date.

A Codelist is defined as a Codelist Template is defined as such on Step 1 of the wizard. Validity Periods may be set on individual items in the same manner as a Codelist defined as “Item Validity”. When the template is submitted to the Registry, the Registry constructs further Codelists from the template. Each of these generated Codelists will be assigned a specific version and will have start and end dates on the Codelist (rather than the individual codes). These generated Codelists are identified in the Fusion Registry as having a type of “generated”. A generated Codelist is not editable or deletable in the Registry and attempting to perform either operation will affect the Template instead.

The following illustrates an example of how Codelist Templates work:

A Codelist Template is created with Agency “SDMX”, ID: “CL_TEMPLATE”. The Registry automatically assigns this a version of 0.1. The Codelist Template refers to 3 codes: Code A with no validity periods; Code B with a validity period of 2000-2010; Code C with a validity period where the start date is 2005 but no end date has been specified.

When the Template is submitted to the Registry, the following structures will be created:

- Final codelist SDMX:CL_TEMPLATE(1.0) – valid up to 2000, contains code A only
- Final codelist SDMX:CL_TEMPLATE(1.1) – valid from 2000 to 2005, contains codes A and B
- Final codelist SDMX CL_TEMPLATE(1.2) – valid from 2005 to 2010, contains codes A, B and C
- Final codelist SDMX:CL_TEMPLATE(2.0) – valid from 2010 onwards, contains codes A and C

The reason that there is a jump in version in the generated Codelists from 1.2 to 2.0 is that the 2.0 Codelist represents a non-backwardly-compatible change.

If the template is modified and then re-submitted, the existing generated templates are removed from the Registry and replaced by the new Template submission.

Deleting a Codelist Template from the Registry, will delete all Codelists that were generated from that Template. Templates may not be referenced by other structures and it is also not possible to convert a Template into another type of Codelist (e.g. Item Validity).