

OIM Taxonomy

Technical Introduction

David Bell AM2

On behalf of the OIM Working Group

The information contained in this presentation represents the opinions of the speaker and may not represent the views of XBRL International, the Board of XBRL International or the consensus opinions of the XBRL Standards Board or Best Practices Board. Nothing in this presentation should be taken to be investment advice and all data representations are merely indicative. The reader should note that only XBRL Specifications that have reached Recommendation status are considered final and suitable for use in software and mission critical systems and must then be used in line with the XBRL International License Agreement. XBRL®, Inline XBRL™, Table Linkbase™ and the XBRL mark are Registered Trademarks or Trademarks of XBRL International Inc in the European Union, United States, China, Japan, India and internationally and may not be used without the permission of XBRL International, Inc including through its Trademark policies and agreements.

Copyright © 2001-2018 XBRL International Inc. All rights reserved. No part of this presentation file may be reproduced in any form or by any means without written permission from XBRL International Inc.

-
- Overview
 - Fundamental Building Blocks
 - Example Taxonomy Walk Through
 - Next steps
-

Simpler, Faster, Better

Goals

Easier, more comprehensible

- Better representation of semantic model
- Syntax independence
- Harmonisation of standards

Designed for modern tools and developers

- XBRL is currently AI resistant – move to models easily consumed by AI tools
- Support use of modern development tools and toolchains
- Remove perception of obsolete technology

Easy to consume

Fit for the future

Overview

Conceptual Model

The specification defines a conceptual object model that is implementation independent

This model consists of objects that encompass both the “taxonomy” and the “instance”

The scope of the model covers features defined within these existing specifications

- XBRL 2.1
- Calculations 1.1
- Dimensions 1.0
- Extensible Enumerations 2.0
- Table Linkbase 1.0

Features from existing specifications are carried across where a clear requirement has been identified

Overview

Principles

The model is composed of cubes ([hypercubes](#))

All facts in a report must be associated with one or more cubes ([hypercubes](#))

The cubes are defined by their dimensions and the dimensions by their domains

Groups ([ELR](#)) are used to associate logically related objects, such as cubes, calculations, presentations and layouts

Evolution not revolution.

Migration must be straightforward

Overview

Principles

XBRL Model objects are the smallest unit of definition ([schema](#), [linkbase](#))

All object definitions must be declared within an XBRL model

There is no explicit distinction between a taxonomy and a report.

A taxonomy is an XBRL model; a report is also an XBRL model.

A report model could include taxonomy definitions (extension elements)

A taxonomy model could include pre-defined facts (tax rates, thresholds, reference data)

Taxonomies and reports combine to form a single XBRL model

Overview

Practicalities

The specification defines an implementation independent model

However, this on its own, is not sufficient for producing interoperable software and conformance suites

Therefore, the specification includes a reference serialization format defined in JSON and accompanied by a corresponding JSON schema

Note that this is still work in progress and is subject to change

Fundamental Building Blocks

The usual suspects

- Abstract
- Concept
- Cube ([Hypercube](#))
- Data Type
- Dimension
- Domain
- Label
- Member
- Reference

And some new ones

- Cube Dimension
- Group ([ELR](#))
- Group Content
- Group Tree
- Layout ([Table](#))
- Network ([Linkbase/Link](#))
- Property ([Attribute](#))
- Relationship ([Arc](#))
- XBRL Model ([Schema](#))

How Does It Work ?

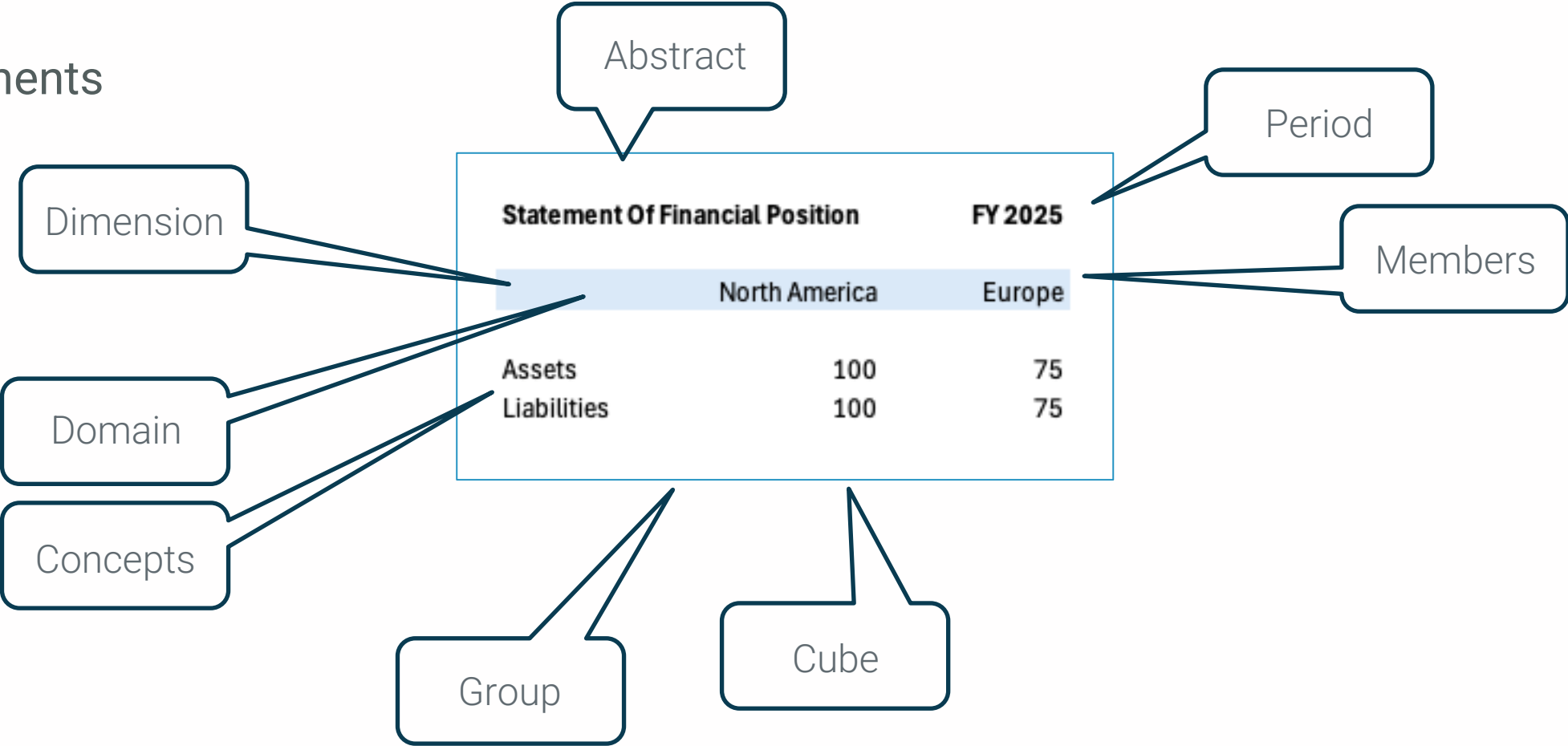
Simple statement

Statement Of Financial Position	FY 2025	
	North America	Europe
Assets	100	75
Liabilities	100	75

How is this taxonomy defined ?

How Does It Work ?

Components



XBRL Model Object

- Is the parent object ([schema](#))
- Has a namespace
- Is named
- Can be labelled
- Will usually import the standard XBRL base definitions model
- All non-nested objects are referenced by name
- All named objects are unique
- Note that for clarity, not all object properties are shown

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	•
abstracts	
concepts	
members	
dimensions	
domainClasses	
domains	
labels	
groups	
groupContents	
cubes	
networks	



ImportedTaxonomies	
xbrlModelName	xbrla:AccountingModel

XBRL Model Object

- The model definitions are accompanied by a Document Info object
- Document Info defines the content type and the namespace declarations
- It can also define the locations of other referenced documents ([schemaLocation](#), [schemaRef](#), [linkbaseRef](#))

```
{
  "documentInfo": {
    "documentType": "https://xbrl.org/2026/model",
    "namespaces": {
      "mini": "http://example.com/mini",
      "xbrl": "https://xbrl.org/2025",
      "xbrla": "http://xbrl.org/accounting"
    },
    "documentNamespace": "mini"
  },
  "xbrlModel": {
    "name": "mini:MiniTaxonomy",
    "frameworkName": "mini-framework",
    "version": "2025",
    "importedTaxonomies": [
      {
        "xbrlModelName": "xbrla:AccountingModel"
      }
    ]
  }
}
```

Abstract Objects

- Separate from concepts
- Used for titles and presentation grouping
- Only have a name
- Can be labelled – as for any named object

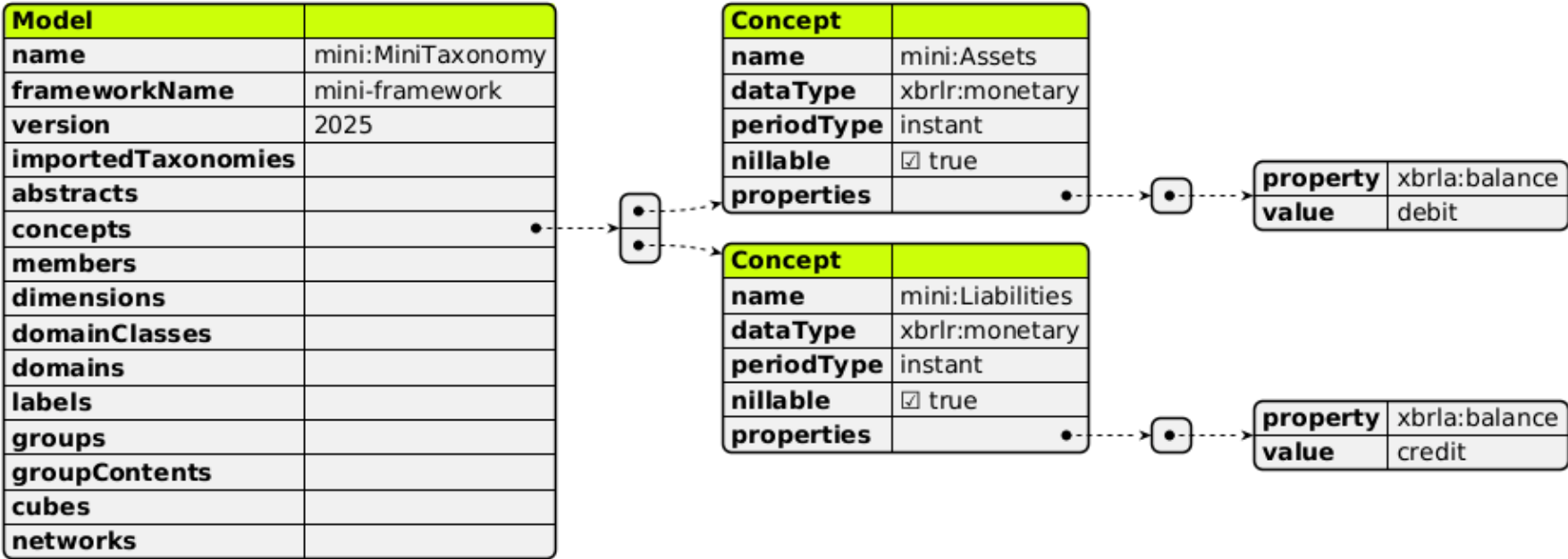
Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	•
concepts	
members	
dimensions	
domainClasses	
domains	
labels	
groups	
groupContents	
cubes	
networks	



Abstract	
name	mini:StatementOfFinancialPosition

Concept Objects

- Usual characteristics
- Balance is defined using properties
- Can be labelled



Concept Objects

- Usual characteristics
- Balance is defined using properties
- Can be labelled

```
"xbrlModel": {  
  "name": "mini:MiniTaxonomy",  
  "frameworkName": "mini-framework",  
  "version": "2025",  
  "concepts": [  
    {  
      "name": "mini:Assets",  
      "dataType": "xbrlr:monetary",  
      "periodType": "instant",  
      "nillable": true,  
      "properties": [  
        { "property": "xbrla:balance", "value": "debit" }  
      ]  
    }  
  ]  
}
```

Label Objects

- Mult-lingual
- Different types (*role*)
- 1:1 relationship from a label to a named object
- Can be used with any named object

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	
dimensions	
domainClasses	
domains	
labels	•
groups	
groupContents	
cubes	
networks	



Label	
relatedName	mini:Assets
labelType	xbrl:label
language	en
value	Assets

Label	
relatedName	mini:Liabilities
labelType	xbrl:label
language	en
value	Liabilities

Member Objects

- Only have a name
- Are untyped
- Can be labelled
- Only for use on dimensions within domains

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	•
dimensions	
domainClasses	
domains	
labels	
groups	
groupContents	
cubes	
networks	



Domain Objects

- Reference an explicit root – defined in a Domain Class ([domain-domainMember](#))
- Member hierarchy ([domainMember – domainMember](#)) is defined using Relationship objects ([arcs](#))
- Domains are not limited to Members – Domains of Concepts can be used for Cubes ([all](#), [notAll](#))

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	
dimensions	
domainClasses	
domains	•
labels	
groups	
groupContents	
cubes	
networks	

Domain	
name	mini:GeographyDomain
root	mini:GeographyRootDomain
relationships	•

Relationship	
source	mini:GeographyRootDomain
target	mini:NorthAmerica

Relationship	
source	mini:GeographyRootDomain
target	mini:Europe

Domain Class Objects

- Is the root object (**domain**) to which domain members can be attached
- Defines the type of domain and allowed values – closed set (**explicit**), open set based on data type (**typed**)
- The Domain is typed – not the Dimension

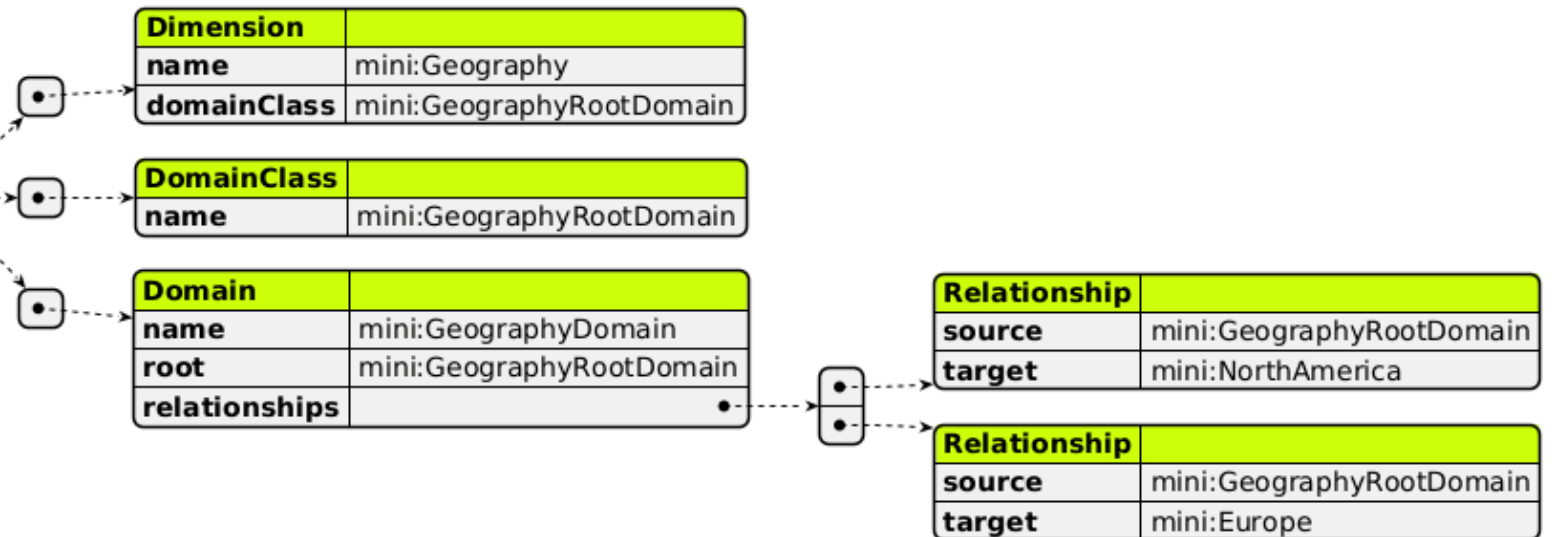
Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	
dimensions	
domainClasses	•
domains	
labels	
groups	
groupContents	
cubes	
networks	



Dimension Objects

- Reference the Domain Class by name ([dimension-domain](#))
- For explicit dimensions, the Dimension can be associated with a Domain in the Cube object definition
- Can be labelled

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	
dimensions	•
domainClasses	•
domains	•
labels	
groups	
groupContents	
cubes	
networks	



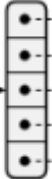
Cube Objects

- Define the appropriate dimensions using Cube Dimension Objects ([hypercube-dimension](#))

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	
dimensions	
domainClasses	
domains	
labels	
groups	
groupContents	
groupContents	
cubes	•
networks	



Cube	
name	mini:balanceSheetCube
cubeType	xbrl:reportCube
cubeDimensions	•



- | Cube Dimension | |
|----------------|--------------|
| dimensionName | xbrl:concept |
- | Cube Dimension | |
|----------------|-------------|
| dimensionName | xbrl:period |
- | Cube Dimension | |
|----------------|----------------------|
| dimensionName | mini:Geography |
| domainName | mini:GeographyDomain |
- | Cube Dimension | |
|----------------|-------------|
| dimensionName | xbrl:entity |
- | Cube Dimension | |
|----------------|-----------|
| dimensionName | xbrl:unit |

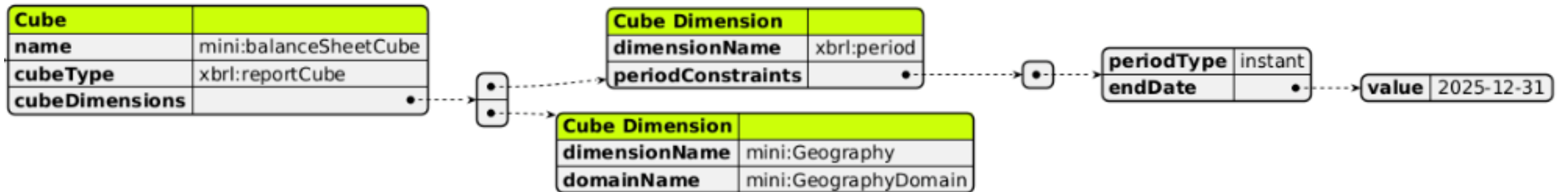
Cube Objects

- Built-in dimensions for concept, entity, period and unit.
- Cube dimensions can also be constrained to limit the facts in the cube (**all**, **notAll**)
- Allows more fine-grained control compared to current specification
- Concept dimension is always included and does not need to be specified if there are no constraints

```
"xbrlModel": {
  "cubes": [
    {
      "name": "mini:balanceSheetCube",
      "cubeType": "xbrl:reportCube",
      "cubeDimensions": [
        { "dimensionName": "xbrl:concept" },
        { "dimensionName": "xbrl:period",
          "domainName": "mini:GeographyDomain"
        },
        { "dimensionName": "mini:Geography" },
        { "dimensionName": "xbrl:entity" },
        { "dimensionName": "xbrl:unit" }
      ]
    }
  ]
}
```

Cube Dimension Objects

- Cube Dimension Objects can define constraints to filter the facts in the cube
- Example period constraint and
- Cube Dimension Objects link the Dimension to the Domain Member network



Network Objects

- Network Objects are used to group relationships ([linkbase](#) / [link](#))
- Allow multiple roots and single objects
- All relationships ([arcs](#)) have the same type - in this case parent - child

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	
dimensions	
domainClasses	
domains	
labels	
groups	
groupContents	
cubes	
networks	•

Network	
name	mini:PresentationNetwork
relationshipTypeName	xbrl:parent-child
roots	•
relationships	•

mini:StatementOfFinancialPosition

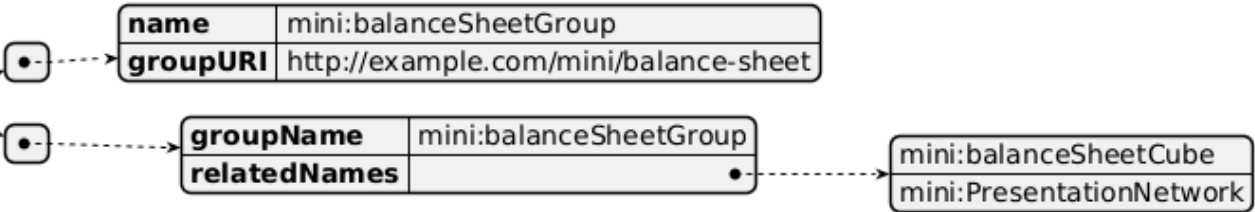
Relationship	
source	mini:StatementOfFinancialPosition
target	mini:Assets
order	1

Relationship	
source	mini:StatementOfFinancialPosition
target	mini:Liabilities
order	2

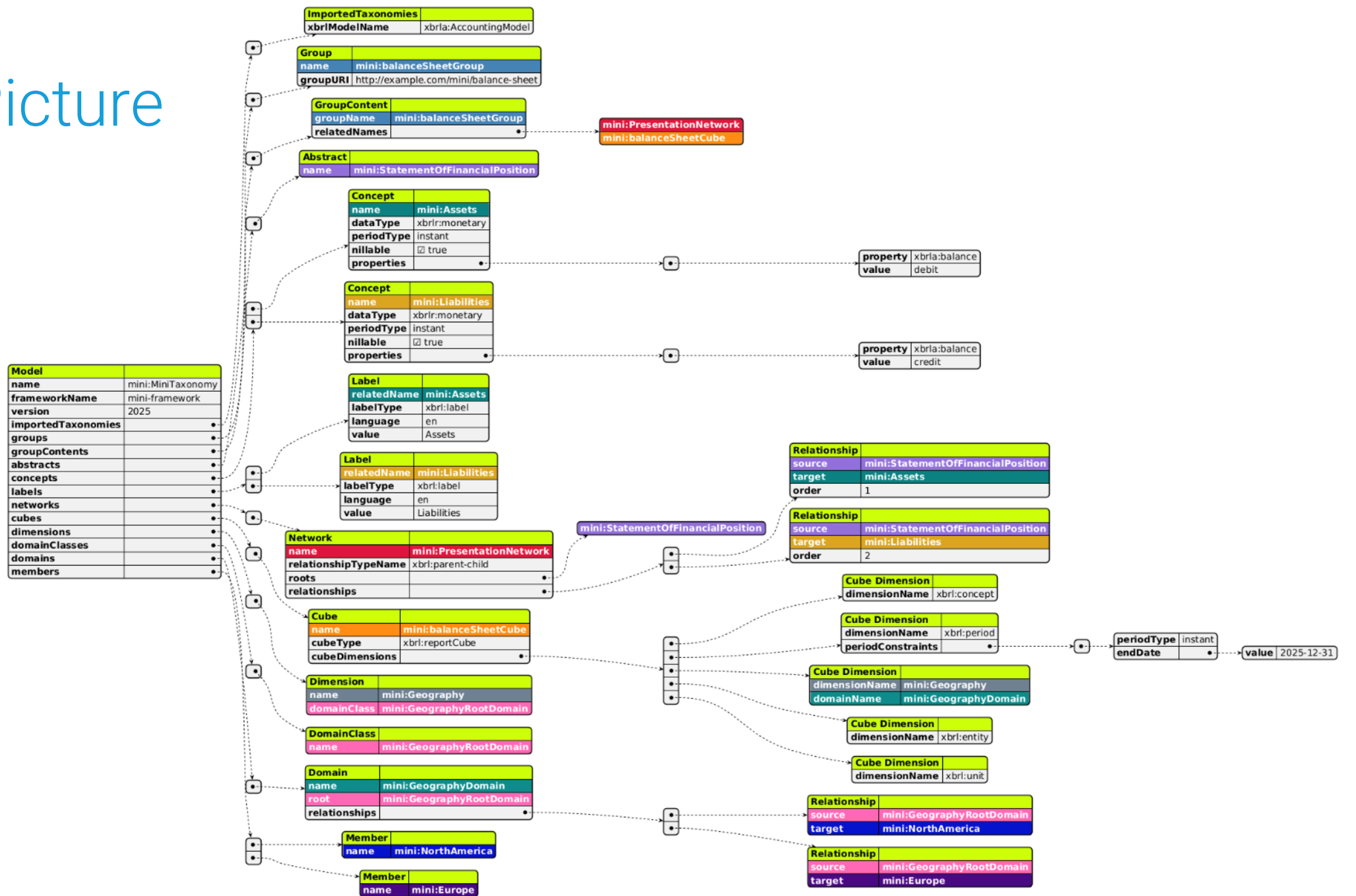
Group Objects

- Group Objects are used to collect related Group Content Objects together ([ELR](#))
- Group Content Objects combine networks, cubes and table templates into coherent groups
- A Group Object can be made up of multiple Group Content Objects with the same groupName

Model	
name	mini:MiniTaxonomy
frameworkName	mini-framework
version	2025
importedTaxonomies	
abstracts	
concepts	
members	
dimensions	
domainClasses	
domains	
labels	
groups	•
groupContents	•
groupContents	
cubes	
networks	



The Big Picture



Recap

Although the example is simple it covers the major components of all current taxonomies

As we have seen it's quite straightforward to build a taxonomy with this model

Taxonomies that use table linkbases may require a bit more work

- The table layout is still work in progress

Could conversion of existing taxonomies be automated ?

- Yes, prototype code for this exists already using a generic approach
- Current thinking in the Working Group is that different taxonomy styles would benefit from different conversion approaches depending upon their construction

Topics We Have Not Talked About

- Data types, measures and units
- Cube types and constraints
- Tables, layouts and forms
- Facts, fact adapters and fact sources
- Extensibility
- Interoperability with XML, CSV and Inline XBRL
- Prototype Taxonomy Viewer

Come to tomorrow's afternoon session for more details

Our **purpose** is to improve the accountability and transparency of business performance globally, by providing the open data exchange standard for business reporting.

Our standards are freely licensed. We operate in the public interest as a not-for-profit global consortium.



Join us
xbrl.org