1. Representing Structures Using Hypercubes

The purpose of this section is to dive into the details related to the structures within an XBRL-based digital financial report. Networks and hypercubes are XBRL technical artifacts that are used for creating structures.

A **hypercube** is a logical scheme that is used to be able to represent sets of information that tends to go together. Reported facts don’t simply float freely in space. Every fact that is provided with in a report is associated with at least one hypercube.

The notion of a hypercube comes from a common model used when working with information, the multidimensional model\(^1\).

The multidimensional model was explained in the chapter, *Introduction to Conceptual Model of a Digital Financial Report*\(^2\), so we will not repeat that here. We do want to remind you of a couple of things however.

**1.1. Important Information to Keep in Mind**

Note that the term hypercube and the term [Table] are synonyms. Likewise, the term dimension, aspect, and [Axis] are synonyms.

A hypercube is an abstract notion which was invented to communicate an idea. Hypercubes can be explicitly defined within the XBRL technical syntax, but an implied hypercube exists when an explicitly defined hypercube is not provided.

In XBRL, there are three core aspects that exist that are always used to describe a fact: entity, period, and concept. Additional noncore aspects can be added to the three core aspects by defining the noncore aspect you need to use within an XBRL taxonomy.

When you represent information within an explicitly defined hypercube, that hypercube has a name and because of that, the hypercube can be used to identify and extract information from within a report. When you define information without an explicitly defined hypercube, i.e. an implied hypercube, you cannot use the name of the hypercube to identify and work with information from a report because there is no physical hypercube that exists, it is only implied therefore it does not have a name.

When the same hypercube is used to represent two different fragments of a report, because each of the two fragments have the same name, you cannot distinguish one fragment from another using the hypercube name (i.e. because it is the same); so you have to also use the network in which the hypercube exists in order to effectively differentiate one hypercube from the other. You can never use a hypercube twice within one network.


XBRL International encourages not mixing dimensional and nondimensional models, i.e. use only explicit hypercubes or use no hypercubes at all. Note that both the US GAAP and IFRS XBRL taxonomies both mix dimensional models.

There tend to be two sound strategies for naming hypercubes. The first strategy is to give every hypercube the same name, say “my:Hypercube”. That approach forces some good modeling practices such as one hypercube per network and also forces software to use one single identifier, the network, to identify a set of information. The other approach is to always use unique, or isomorphic, hypercubes meaning that each named hypercube represents one single set of information.

2. Contrasting Hypercube Structures

In this section I will help you understand hypercubes by comparing and contrasting a number of different hypercubes. All of these hypercubes are 100% the same in terms of the meaning conveyed by the information. The only difference is the approach used to represent the hypercube. The objective is to make the differences between each of these approaches clear. You can download the XBRL files and examine them.

2.1. Implanted Hypercube (i.e. no explicit hypercube defined)

To start, consider a fragment of a financial report that is represented without using a hypercube at all. When no hypercube is explicitly defined there exists one implied hypercube.

Notice at the top of the image where the name of “Table” is indicated to be “Implied [Table]”. Essentially, if a set of facts that exist within one network does not have an explicitly defined hypercube; then you can assume that each of those facts exists in one single implied hypercube which we have simply given the name “ImpliedTable”.

---

5 Implanted hypercube example, http://xbrlslide.azurewebsites.net/2020/Examples/Hypercubes/01-ImpliedHypercube/RollUp-SampleInstance.xml
Representing information without an explicit hypercube within a financial report is not considered a best practice. Explicit hypercubes are the best way to go.

### 2.2. Explicit Hypercube with no Dimensions

In this example we have a fragment of a report that has been represented using an explicit hypercube labeled “Property, Plant and Equipment, by Component [Table]”. However, no additional noncore dimensions were defined.

You can see that in the first example what was called “Implied [Table]” is now called “Property, Plant and Equipment, by Component [Table]”. As such, you can now identify the set of facts using the hypercube that contains the set of facts by its name. Further, there is nothing wrong with representing a set of information using a hypercube but not associating any additional noncore dimensions with that hypercube. That is perfectly legitimate per the XBRL technical specification.

### 2.3. Explicit Hypercube, One Noncore Dimension, no Dimension Default

In this example we have a fragment of a report that has been represented using an explicit hypercube labeled “Property, Plant and Equipment, by Component [Table]”. In addition, we added the dimension “Legal Entity [Axis]” and one member of that dimensions domain, “Consolidated Entity [Member]”.

---

6 Explicit hypercube with no dimensions example, [http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/02-ExplicitHypercubeNoAxis/RollUp-SampleInstance.xml](http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/02-ExplicitHypercubeNoAxis/RollUp-SampleInstance.xml)

7 Explicit hypercube, one noncore dimension, no default dimension example, [http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/03-ExplicitHypercubeWithAxisNoDefault/RollUp-SampleInstance.xml](http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/03-ExplicitHypercubeWithAxisNoDefault/RollUp-SampleInstance.xml)
In this representation approach, the legal entity is explicitly identified as being the consolidated reporting entity. Whereas in the first and second representations the reader of the information and software would have to imply that the information is related to the consolidated entity.

One important point being made by this representation is that the notion of a dimension default is not used to represent the “Consolidated Entity [Member].” As such, the dimensional information physically exists within the XBRL instance which you can see if you examine the XBRL of the XBRL instance:

8 Explicit hypercube, one noncore dimension, with a dimension default, http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/04-ExplicitHypercubeWithAxisWithDefault/RollUp-SampleInstance.xml

The next example will make the notion of a dimension default and its purpose clear.

2.4. **Explicit Hypercube, One Noncore Dimension, With a Dimension Default**

In this example the only thing that has changed is that we added a dimension default to the noncore dimension that has been provided “Legal Entity [Axis]”. What adding the dimension default achieves is making the representation more flexible^8^.
Suppose that you also wanted to represent the line item “Property, Plant and Equipment, Net” on the balance sheet. If you look at the XBRL instance of the example above, the dimension information is explicitly provided within the context of the fact as we have pointed out. However, in the XBRL instance of this example, the dimensional information does not explicitly appear within the XBRL instance context.

You might be asking yourself, why does this matter? It matters because many times one fact appears in two different fragments of a report. For example, you see the breakdown of the subcomponents of property, plant, and equipment provided in this fragment. You can imagine that the total of this breakdown of subcomponents would also appear, say, as a line item on the balance sheet.

What a dimension default does is make it possible to use the same fact in two different fragments of a report rather than having to provide duplicate facts, one for each fragment, when in reality you only have one fact. That is the role of a dimension default, to enable the interconnections between hypercubes of information.

2.5. Explicit Hypercube, Adding Additional Members, Including Dimensional Aggregation

In this example we add two additional members to the single noncore dimension “Legal Entity [Axis]”. The members added are “Business Unit Alpha [Member]” and “Business Unit Bravo [Member].” Further, the sum of each business unit aggregates to the consolidated entity. That aggregation across members of the dimension need to be validated using an XBRL Formula as XBRL calculations cannot validate information across the members of a dimension. What adding this information achieves is provide additional details for the reported facts which disaggregates the information by business unit9.

9 Explicit hypercube, adding additional members, including dimensional aggregation example, http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/05-ExplicitHypercubeWithAggregation/RollUp-SampleInstance.xml
2.6. **Explicit Hypercube, Adding Second Noncore Dimension**

In this example we have added a second noncore dimension “Scenario [Axis]” that has a single member “Actual [Member]”. What adding this information achieves is that it makes explicit that the information provided within this fragment is “actual” as contrast to “forecasted” information.

2.7. **Summary**

Logically, each and every representation shown above is the same except for the fact that the last two representations provide additional detailed facts. But the representation approaches used where each slightly different.

The number one takeaway you should have is a conscious understanding of the differences. Which representation approach to use depends on your reporting situation. All things considered, the following is always good advise and the best practice approach to representing information:

1. Always use explicit uniquely named hypercubes and have every fact reported within at least one hypercube. Both the SEC, ESMA, and XBRL international\(^\text{11}\) suggest this approach.

\(^{10}\) Explicit hypercube, adding second noncore dimension example, [http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/06-ExplicitHypercubeWithAggregationTwoAxis/RollUp-SampleInstance.xml](http://xbrlsite.azurewebsites.net/2020/Examples/Hypercubes/06-ExplicitHypercubeWithAggregationTwoAxis/RollUp-SampleInstance.xml)

2. Always provide a dimension default for every dimension.
3. Only provide noncore dimensions to the extent that they are helpful in making facts distinguishing from one another.
4. Avoid adding unnecessary, meaningless, or ambiguous hierarchy within your line items concepts or dimensions members.

Note that any additional dimensions or line items can be included within a hypercube and all of these representation patterns apply.

3. Structure Representation Strategy

In this section I help you understand the different structure representation strategies, understand the pros and cons of each, and help you pick structure representation strategy that works best for you for your situation.

There are two fundamental ways to identify a structure using a machine-based process:

1. Using a unique identifier: Pointing to some unique identifier that provides a “handle” which you can use to “grab” that thing.
2. Prototype theory: Looking at the specification or description of some thing and using that specification/description to examine the properties of some set of things to determine if a specific thing is what you are looking for. (i.e. undisputed example)

As an example, one can understand that something is a “chair” by (#1) seeing a label “chair” and identifying that thing as being a chair; or (#2) understanding as many properties as possible about what a “chair” is; looking at the properties of a thing until the “chair” is identified and given no other instance or indicator that there is something better that meets the specification better; you can reliably conclude that what you have found using the properties is likely the chair (or other thing) you have been looking for.

Using XBRL there are two technical artifacts that can be used to represent structures: Networks and Hypercubes. There are two things that drive whether you use a Network and/or Hypercube to define something:

1. Because you have to in order to prevent a technical conflict.
2. Because you want to in order to organize something in a particular way.

Given the two ways to identify structures and the two ways to represent structures, there seems to be essentially four approaches or strategies for representing structures effectively:

1. Hypercube as Unique Identifier: Explicitly define a unique hypercube to identify each structure.
2. Network as Unique Identifier: Explicitly define a unique network to identify each structure (with all structures using the same hypercube).
3. Disclosure Specification of Prototype: Explicitly define a specification to describe each structure (with neither a hypercube nor a network explicitly describing the structure).
4. **Hypercube as Unique Identifier Plus Disclosure Specification of Prototype:**
Explicitly define a structure using *a unique hypercube and also define a specification* for each structure.

5. **Network as Unique Identifier Plus Disclosure Specification of Prototype:**
Explicitly define a structure using *a unique network and also define a specification* for each structure.

There are other approaches that can be used, but the approaches will not work to let you identify the structure or determine if the structure is represented correctly. I will go over the first four approaches that work first; then show you several approaches that do not work in order to contrast approaches that do work and approaches that will not work.

### 3.1. **Hypercube as Unique Identifier**

Using this approach, each hypercube is given a unique name and therefore the unique name can be used to identify the structure:\textsuperscript{12}

<table>
<thead>
<tr>
<th>Component: (Network and Table)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Network</strong></td>
</tr>
<tr>
<td>0.2-Comprehensive Income</td>
</tr>
<tr>
<td>(<a href="http://www.xbrlsite.com/fdsr5/role/ComprehensiveIncome">http://www.xbrlsite.com/fdsr5/role/ComprehensiveIncome</a>)</td>
</tr>
<tr>
<td><strong>Table</strong></td>
</tr>
<tr>
<td>Comprehensive Income Statement [Hypercube]</td>
</tr>
</tbody>
</table>

Slices (applies to each fact value in each table cell)

| Reporting Entity [Axis] | GH25940D0F8BUJ9L56S11 (http://standards.iso.org/iso/17442) |

Given that each hypercube is unique and can therefore be used to reliably identify the structure and given that the structure is considered to be an undisputed example of the disclosure, there is no need to provide disclosure mechanics rules because there is enough information in the structure to provide a specification for evaluating if other structures created are consistent with the provided prototype.

### 3.2. **Network as Unique Identifier**

Using this approach, every hypercube has exactly the same name and therefore it cannot be used to identify a structure; however, the network has a unique name and therefore it can be used as a unique identifier:\textsuperscript{13}

---

\textsuperscript{12} Example of approach 1, Hypercube as Unique Identifier, [http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach1/](http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach1/)

\textsuperscript{13} Example of approach 2, Network as Unique Identifier, [http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach2/](http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach2/)
Note that this will only work if those creating XBRL instances are required to use the network name from the base taxonomy within their report and/or extension taxonomy. Just like the first case, no disclosure mechanics rules are necessary because the base taxonomy disclosure is a prototype specification and undisputed example and therefore can be used to evaluate reports to see if the disclosure mechanics of a report is consistent with the prototypes provided in the base taxonomy.

3.3. Disclosure Specification as Prototype

Using this approach, there is no reliable unique identifier for the hypercube or the network; however, a specification is provided that indicates the characteristics of a disclosure and provides a unique name for each disclosure\(^\text{14}\):

Even though there are no unique names for either hypercubes or networks, the disclosure can be identified using a specification of the disclosure mechanics rules and a list of uniquely named disclosures:

In order to uniquely identify and be able to refer to each structure, a unique disclosure name must be created otherwise there is no way to refer to a disclosure because there are no unique network names or hypercube names that can be referred to. In this case, a separate XBRL taxonomy schema\(^\text{15}\) is created to name disclosures. This can be done as part of the creation of a base taxonomy or supplemental to the base taxonomy after it has been created:

3.4. Hypercube as Unique Identifier Plus Disclosure Specification of Prototype

Using this approach is essentially a combination of approach #1 and approach #3 so you have both unique hypercubes that can be used to identify structures and the disclosures they represent and specifications that describe the disclosures\(^\text{16}\):

\(^{15}\) XBRL taxonomy schema with uniquely named disclosures, http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach3/disclosures.xsd

You see above that the comprehensive income statement\(^{17}\) can be reliably identified using the hypercube used to represent that structure. The same is true for the balance sheet structure\(^{18}\) and changes in equity structure\(^{19}\).

As can be seen per the disclosure mechanics rules, the unique hypercube name is used to help identify the disclosure in the disclosure mechanics rules. In addition, there is information such as the information model (i.e. roll up, roll forward), the total concept of the roll up or instance concept of a roll forward, or other information that is useful in specifying or identifying a structure which represents some specific disclosure. However, if you either (a) remove all information other than the name of the hypercube or (b) use only the name of the hypercube; you get the same result.

Note that in approach #3’s disclosure mechanics rules hypercubes are not used.


\(^{19}\) Human readable, changes in equity structure, [http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach4/evidence-package/contents/index.html#Rendering-ChangesInEquity-sfac6_ChangesInEquityHypercube.html](http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach4/evidence-package/contents/index.html#Rendering-ChangesInEquity-sfac6_ChangesInEquityHypercube.html)
As such, providing both a unique hypercube plus the disclosure mechanics rules seems redundant or overkill. However, this is only the case if the base taxonomy structures can be used as undisputed examples of a specification of a disclosure.

3.5. **Network as Unique Identifier Plus Disclosure Specification of Prototype**

Using this approach is essentially a combination of approach #2 and approach #3 so you have both unique hypercubes that can be used to identify structures and the disclosures they represent and specifications that describe the disclosures:\(^{20}\):

![Hypercube Diagram](image)

In addition to the unique network identifier, disclosure mechanics rules are provided however in this case the disclosure mechanics rules do not leverage the unique network identifier to identify the disclosure that the structure is representing:\(^{21}\):

---


As such, providing both a uniquely identifiable structure plus the disclosure mechanics rules in order to identify the disclosure appears redundant or overkill. However, this is can only be the case if (a) networks from the base taxonomy are used in reports and (b) the base taxonomy can be views as an undisputed example of the disclosure.

### 3.6. Pathological Example Provided for Contrast

Note that this is a pathological example provided for contrast and not considered a viable approach to representing structures that are used to provide disclosures within an XBRL-based report. Note the following structure:\(^{22}\):

Assume that the reporting economic entity created their own network identifiers similar to XBRL-based financial statements submitted to the SEC.

How exactly would you refer to this structure within an XBRL-based digital financial report? Note that the network identifier would be unique for a reporting economic entity, but every economic entity would provide a different unique network identifier for this or any other structure. Further, there is no hypercube at all and so that cannot be leveraged to identify the structure. If the reporting economic entity used an extension hypercube, that would not be helpful in identifying the structure.

\(^{22}\) Approach 6, Pathological Example Provided for Contrast, [http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach6/](http://xbrlsite.azurewebsites.net/2020/structure-representation-strategy/Approach6/)
Suppose you wanted to write rules that specify what each of the three disclosures looked like in order to control a machine-based process; the balance sheet, comprehensive income, and changes in equity. Suppose you wanted to, say, create the rules below:

<table>
<thead>
<tr>
<th>#</th>
<th>Disclosure</th>
<th>Date</th>
<th>Level</th>
<th>Pattern</th>
<th>Disclosure</th>
<th>Disclosure Consistent</th>
<th>Applicable</th>
<th>Representation Concept [TEXT BLOCK]</th>
<th>Representation Concept DETAIL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Balance Sheet</td>
<td>Jan.</td>
<td>LevelDetail</td>
<td>Hierarchy</td>
<td>True</td>
<td>CONSISTENT</td>
<td>True</td>
<td>NOT-EXPECTED</td>
<td>Assets</td>
</tr>
<tr>
<td>2</td>
<td>Changes in Equity</td>
<td>Jan.</td>
<td>LevelDetail</td>
<td>Rollup</td>
<td>True</td>
<td>CONSISTENT</td>
<td>True</td>
<td>NOT-EXPECTED</td>
<td>Equity</td>
</tr>
<tr>
<td>3</td>
<td>Comprehensive Income</td>
<td>Jan.</td>
<td>LevelDetail</td>
<td>Rollup</td>
<td>True</td>
<td>CONSISTENT</td>
<td>True</td>
<td>NOT-EXPECTED</td>
<td>Comprehensive Income</td>
</tr>
</tbody>
</table>

Note that the rules above refer to a disclosure: “disclosures:BalanceSheet”, “disclosures:ChangesInEquity”, “disclosures:ComprehensiveIncome”. But you have nothing of the sort to refer to in this pathological example, it has not been defined.