Understanding Networks, Hypercubes, Components, and Information Blocks

by

Charles Hoffman, CPA

Helpful insights into the pros and cons of representing information using XBRL.

2024-01-06 (DRAFT; Work in Progress)

https://digitalfinancialreporting.blogspot.com/2024/12/master-class-in-representing-financial.html

1. Introduction

This document compares and contrasts ten XBRL-based financial statements which contain exactly the same financial report logic but where represented within the XBRL technical syntax using different approaches.

Each of the technical syntax approaches is completely valid, and pass XBRL technical syntax validation per any fully compliant XBRL processor. Further, the logic of the reported financial information is exactly the same in each of the ten financial statements.

What is different is the approach used to represent the financial logic within the XBRL technical syntax.

By understanding and studying the similarities and differences between these ten representations one can better understand good practices, best practices, and poor practices for representing financial logic within an XBRL-based financial statement.

To help explain the details which will be explained in this document and to keep that explanation as easy as possible; XBRL-based reports that are both very sophisticated by as small as possible are utilized. As far as the author knows, 100% of the possible logic that is representable within XBRL is covered by these examples.

1.1. Brief Overview of Examples

The following is a very brief overview of the example XBRL-based reports that will be compared and contrasted. Complete versions of each of these reports is provided in several different forms. Note that the actual names of each example are meaningless and provide only enough to be able to understand what example I am looking at.

1. **PROOF-Master**[1]: This is the baseline. This has pretty much all the logical complexity that anyone creating an XBRL-based digital financial statement would every have to get their heads around and deal with. Information is represented very consistently and with one explicit hypercube within a network.
2. **PROOF-Alternative1**[2]: This takes the baseline (the Master) and reorganizes the blocks of information differently within the XBRL networks and hypercubes. As many networks were removed as possible. This has one subtle issue; the first hypercube has many hypercubes but no "root" to organize those hypercubes.
3. **PROOF-Alternative2**[3]: This is exactly the same as Alternative1, except if you look in the first network, there is now a "root" or "container" that is used to organize the many hypercubes in the first network.

[1] PROOF-Master, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-master/index.html
[2] PROOF-Alternative1, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-alternative1/index.html
[3] PROOF-Alternative2, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-alternative2/index.html

4. **PROOF-Hypercubes2**[4]: This is exactly the same as the Master or baseline, except that rather than giving hypercubes unique names, every hypercube is expressed using one standard hypercube named "Standard [Hypercube]".
5. **PROOF-Hypercubes3**[5]: This is exactly the same as Hypercube2 except that both "Standard [Hypercube]" and "Hypercube [Line Items]" are used.
6. **PROOF-Implied**[6]: This is exactly the same as the Master or baseline except that explicit hypercubes where only used when they are required because noncore dimensions must be used to represent a financial disclosure.
7. **PROOF-Dimensions**[7]: This is exactly the same as the Master or baseline except that every hypercube explicitly defines every dimension used anywhere in the financial statement.
8. **PROOF-Blocks**[8]: This is exactly the same as the Master or baseline except that every possible [Abstract] report element, used to organize the presentation relations, was removed.
9. **PROOF-Sparse**[9]: This is exactly the same as the Alternative2 except that a hypercube that represents a segment breakdown is intermingled with multiple other blocks of information causing a "sparse" hypercube as a result (i.e. a hypercube with a lot of blank facts because of a non-best practices use of hypercubes).
10. **PROOF-Proper**[10]: This is very similar to the Master or baseline except that duplicate hypercubes where made unique and other issues were fixed.

All the examples can be viewed online in various forms, downloaded for further inspection, loaded into the XBRL software application of your choice.

---

[4] PROOF-Hypercube2, http://www.xbrlsite.com/site1/seattlemethod/platinum-testcases/proof-hypercubes2/index.html
[5] PROOF-Hypercube3, http://www.xbrlsite.com/site1/seattlemethod/platinum-testcases/proof-hypercubes3/index.html
[6] PROOF-Implied, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-implied/index.html
[7] PROOF-Dimensions, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-dimensions/index.html
[8] PROOF-Blocks, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-blocks/index.html
[9] PROOF-Sparse, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-sparse/index.html
[10] PROOF-Proper, http://www.xbrlsite.com/seattlemethod/platinum-testcases/proof-proper/index.html

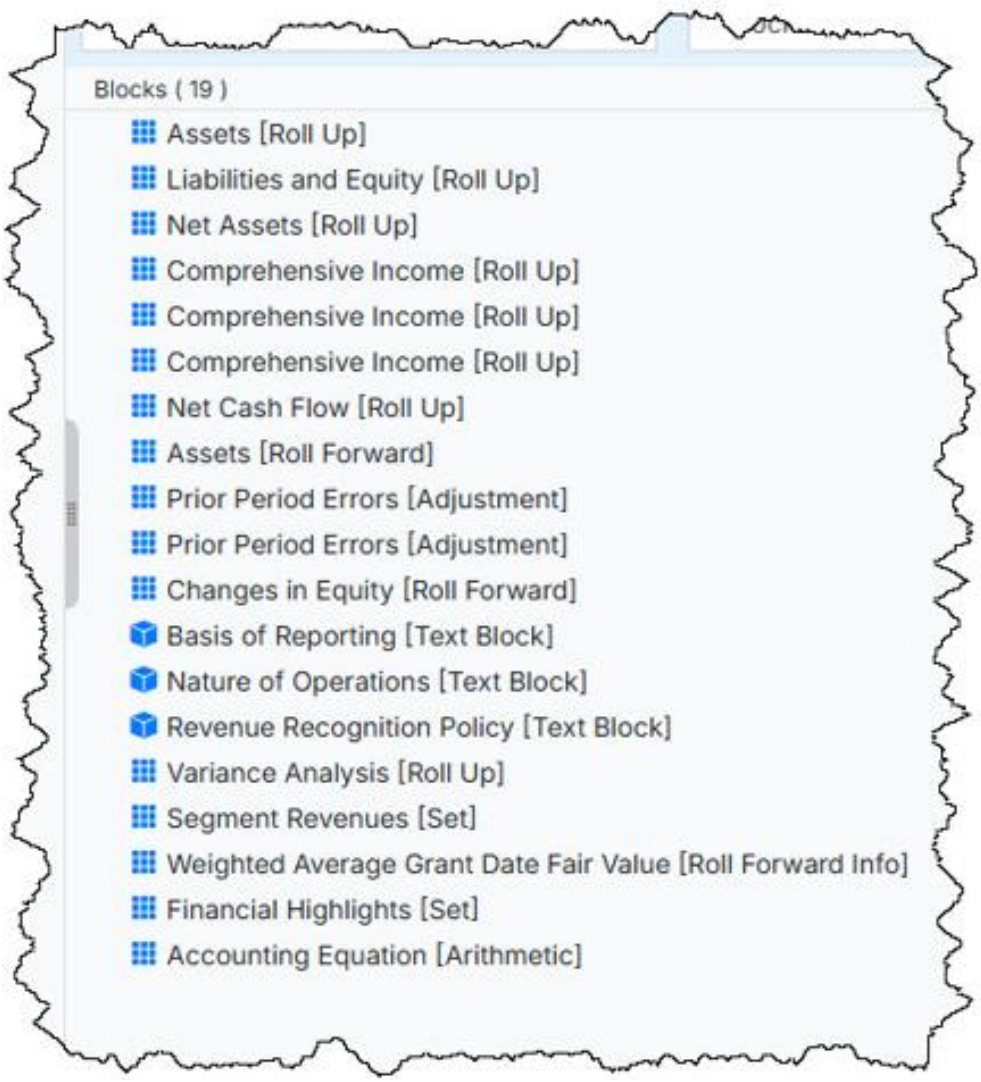
2. Comparing and Contrasting Examples

In this section, each example is explained and compared/contrasted to another example and the incremental change between the two examples is discussed.

Each of the individual representations in XBRL provides exactly the same logical information in terms of the blocks and disclosures of financial logic provided.

Within each of the ten reports there are exactly 19 identifiable[11] blocks of information. There blocks of information can be identified using software-based processes. Every piece of information conveyed by an XBRL-based report must exist within a block of information, which exists within a hypercube (explicitly defined or implied), which exists within a network.

These are the **blocks of information**, 19 of them, which exist in each of the ten XBRL-based representations:

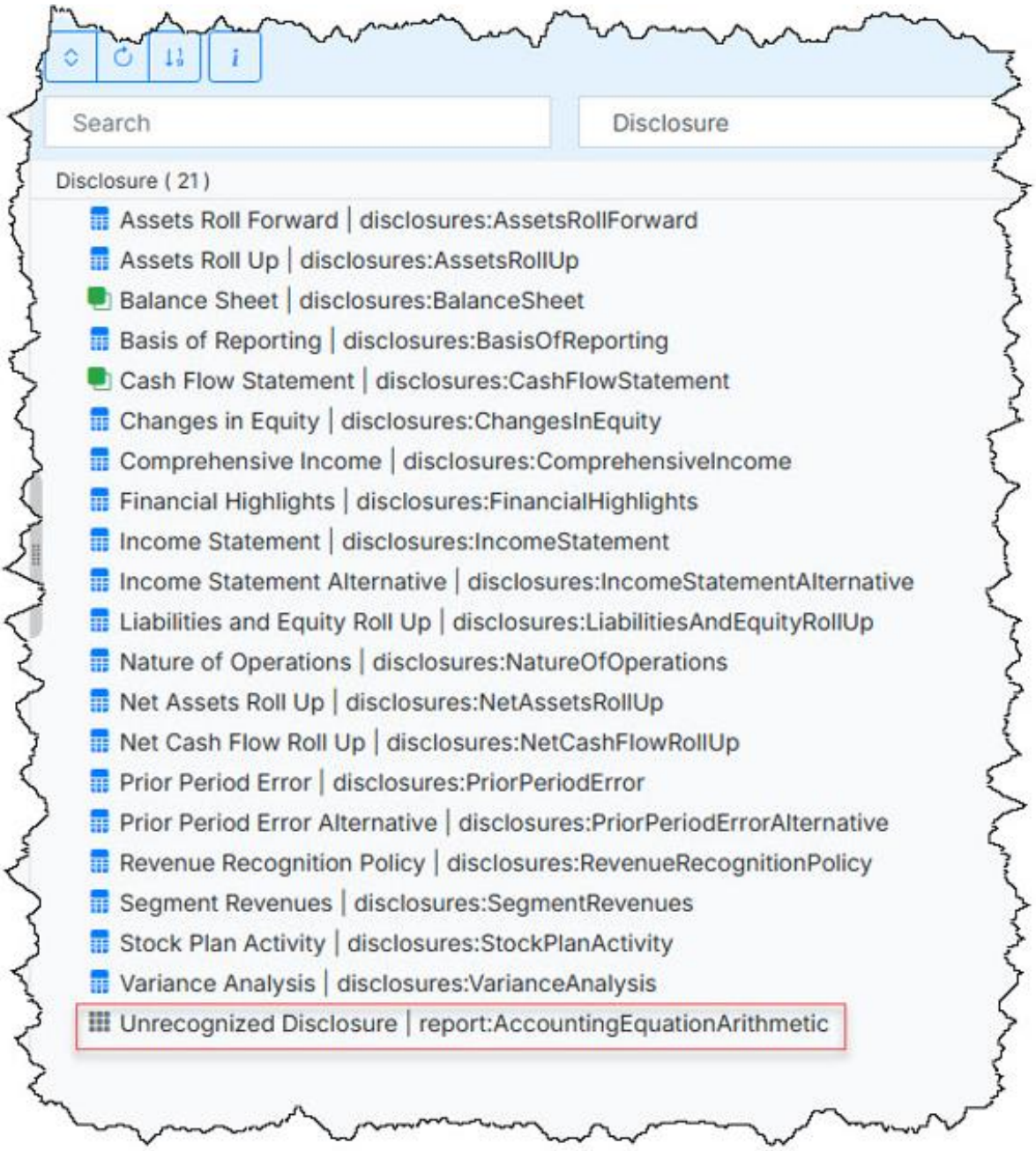


[11] *Information Model Identification*, http://www.xbrlsite.com/mastering/InformationModelIdentification.pdf

Every block of information is identifiable as one or more of the information logic patterns. The organization of members of a dimension has logic patterns and the organization of the concepts and abstracts within a set of LineItems has logic patterns.

In addition, if (a) information is provided within a base XBRL taxonomy, (b) by a supplemental XBRL taxonomy; then every block of information can be further identified as being a specific disclosure.

The following are the disclosures which are contained within each of the ten example representations:



A disclosure is defined simply as something that is disclosed within a report, be that disclosure part of the primary financial statements, part of the policies, or part of the disclosure notes of a financial statement.

Note that there are 21 disclosures but only 19 information blocks. This is because the Balance Sheet disclosure is made up of two information blocks (Assets [Roll Up] and Liabilities and Equity [Roll Up] and the Cash Flow Statement disclosure is likewise made up of two information blocks (Net Cash Flow [Roll Up] and Assets [Roll Forward]). Basically, the Balance Sheet disclosure and Cash Flow Statement Disclosure information blocks appear twice; both separately and as combined.

Effectively, those blocks of information and disclosures are organized differently within each of the ten examples. All information is organized within networks; individual representation could put the blocks of information within one network or within some other network. All information could be in one hypercube, in another hypercube, or in implied hypercubes if no noncore dimensions are used in the representation. Noncore dimensions may, or may not, be provided where they are not necessarily required to be provided. Abstract concepts used to organize the report model may, or may not be provided when there is an alternative. Names of hypercubes may be the same or be different. The name of the [Line Items] type of report elements may, or may not, be the same.

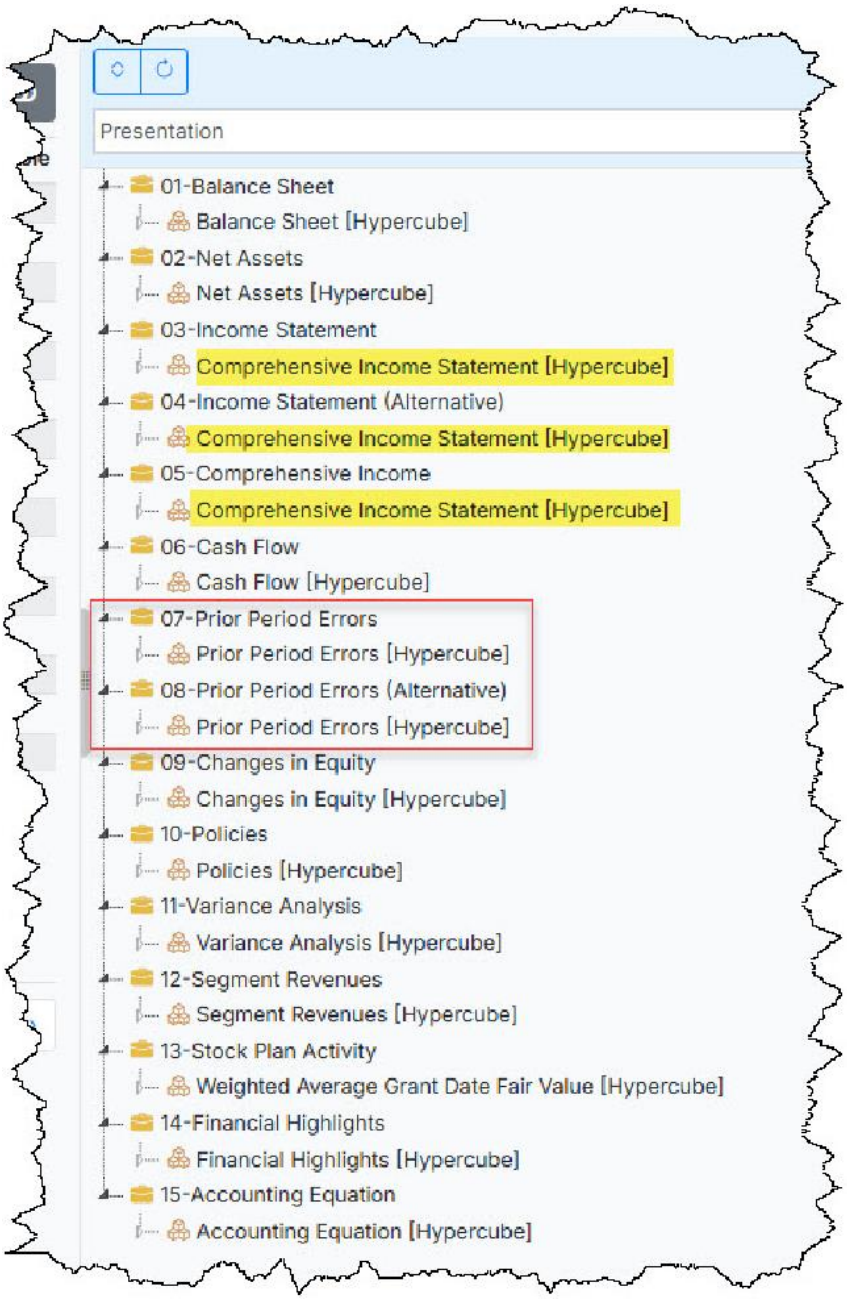
All this will be shown and explained in each of the ten examples which we will get into now.

2.1. PROOF-Master

This is the baseline. This is effectively my PROOF[12] which is used in testing software, testing and experimenting representing information within an XBRL-based report to make sure it acts correctly and interacts with other representations correctly.

This has pretty much all the logical complexity that anyone creating an XBRL-based digital financial statement would every have to get their heads around and deal with. Information is represented very consistently and with one explicit hypercube within a network. The PROOF is intentionally not perfect because it has to represent how reports are actually being created in the real world. However, the PROOF does need to be complete, consistent, and precise. The following graph provides the networks and hypercubes defined within the PROOF-Master:

[12] PROOF, https://digitalfinancialreporting.blogspot.com/2023/12/proof.html



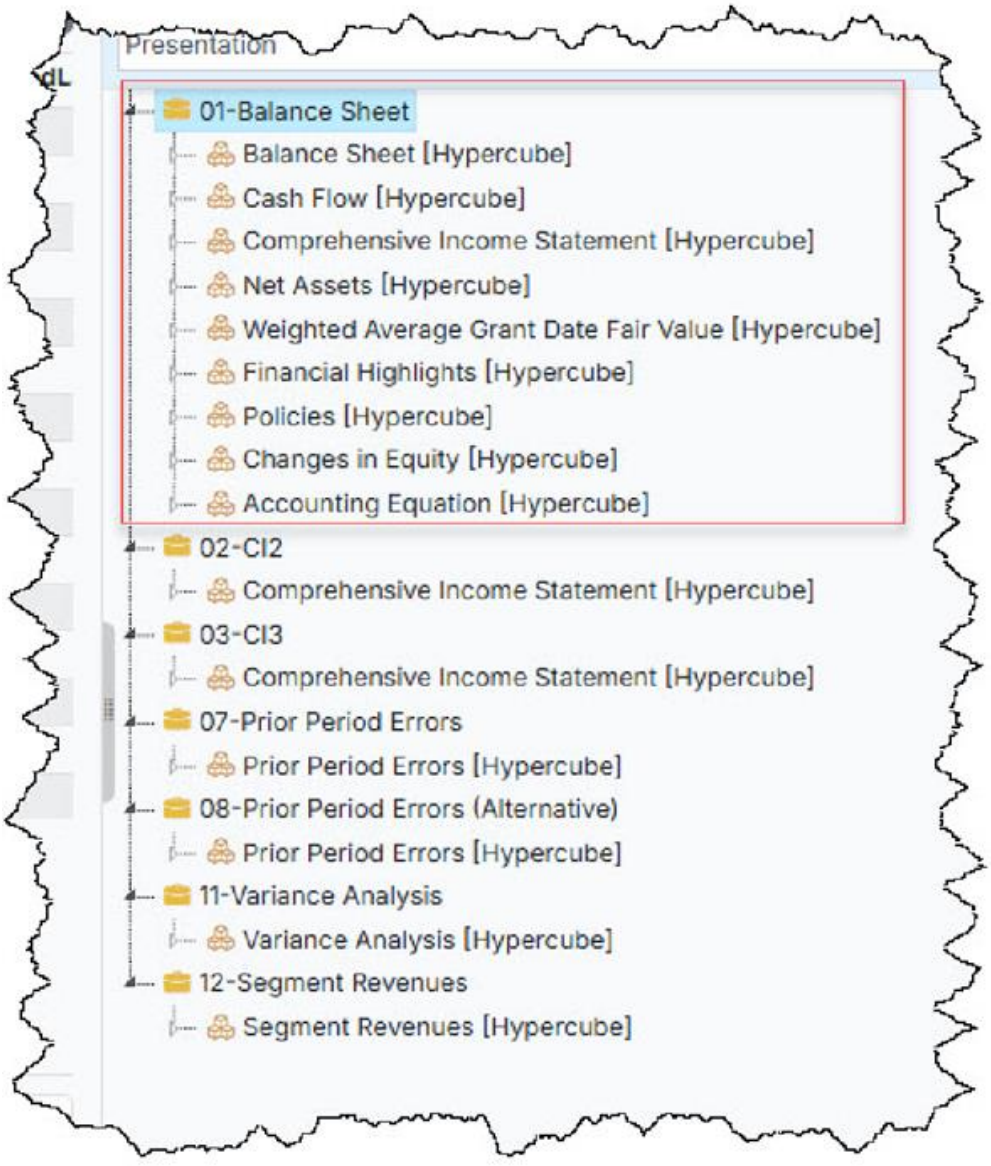
Notice that some information is being represented using the same hypercube, "Comprehensive Income Statement [Hypercube]" and "Prior Period Errors [Hypercube]". Notice that it is hard to know exactly what block of information exists within the hypercubes that have the same name. If you compare the PROOF-Master graphic with the PROOF-Proper graphic which fixes that use of the same hypercube name to represent different information blocks, you can see the pros and cons of polymorphic hypercube as contrast to isomorphic hypercubes.

Identifying specific information in a report is easier if isomorphic hypercubes are used, meaning every hypercube name is used to represent a unique information artifact.

2.2. PROOF-Alternative1

This example takes the baseline (the Master) and reorganizes the blocks of information differently within the XBRL networks and hypercubes. As many networks were removed as possible. This has one subtle issue; the first hypercube has many hypercubes but no "root" to organize those hypercubes.

Note that there are now only seven networks, rather than 15 in PROOF-Master, because a number of hypercubes were modeled within the same network as contrast to being represented in their own network.



Note that again, there are 15 hypercubes. That this example shows that it is very possible to represent some hypercubes in different networks; however, in other cases is would be impossible to combined certain specific hypercubes because conflicts would occur between hypercube information. In other cases it is impossible to combined certain specific hypercubes in the same network because things like XBRL calculation

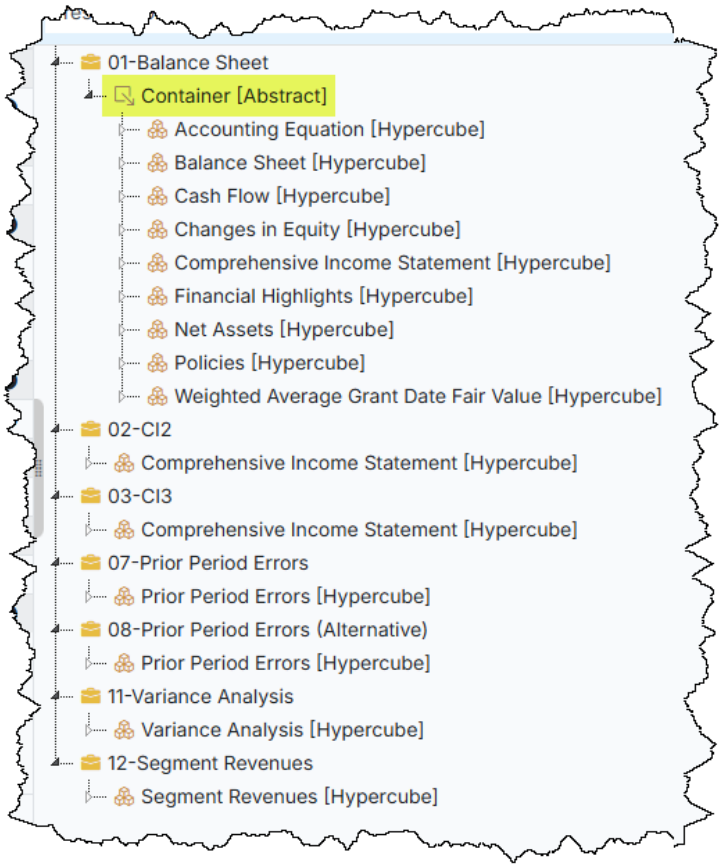
relation rules would cause conflicts even though there would be no conflict between the dimensional information.

There is one issue with the PROOF-Alternative1 representation which will be shown by using the PROOF-Alternative2 representation. Note that the first network has no way to organize the nine hypercubes that exist within that first network. The order of the hypercubes is arbitrary.

2.3. PROOF-Alternative2

This example is exactly the same as Alternative1, except that if you look in the first network in this example, there is now a "root" or "container" that is used to organize the many hypercubes in the first network.

Here you see the concept "Container [Abstract]" whose sole purpose is to enable the nine hypercubes within the "01-Balance Sheet" network to be put into a desired specific order:

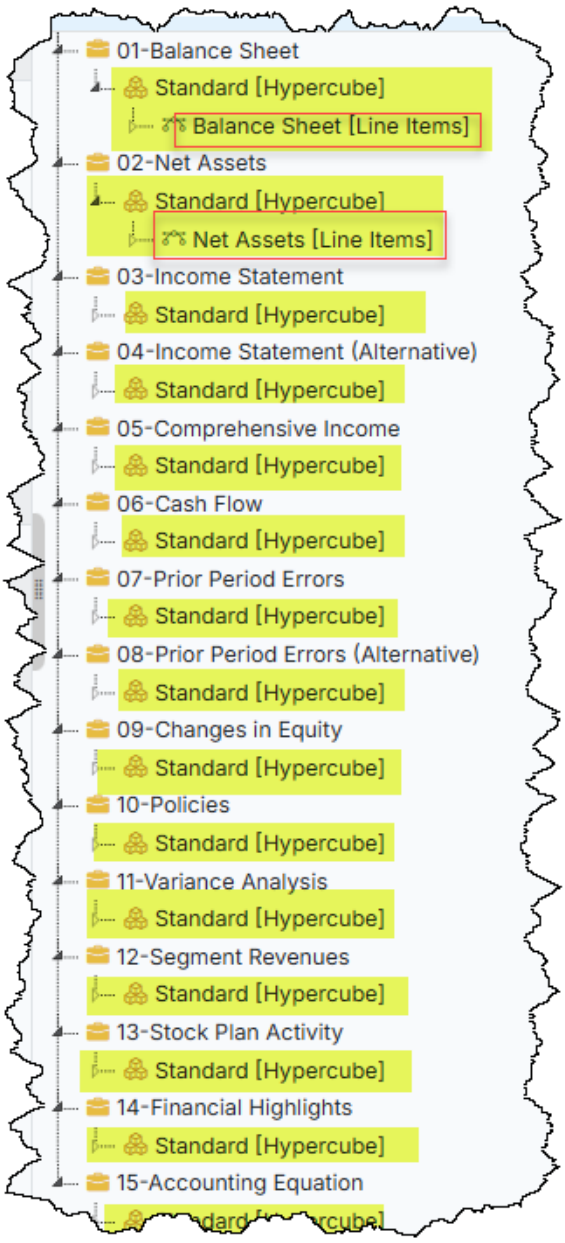


Nothing else is different. There are three key points here. First, it is very possible to put multiple hypercubes into the same network. Second, it is not possible to put any hypercube into any network; sometimes you are constrained by the type of information represented within a hypercube as to what For example, XBRL calculation relations and XBRL Dimensions work via completely different sets of rules. Hypercubes do not constrain XBRL calculation relations; networks do. Also how information is rendered for human readable consumption and how those renderings are shown within software applications is also a consideration.

This brings up a consideration: what exactly would be the reasoning behind not having one hypercube per network? What advantage is there. There is an advantage to consistently having one network contain one hypercube; consistency. With one network containing one hypercube conflicts can be avoided.

2.4. PROOF-Hypercubes2

This example is exactly the same as the Master or baseline, except that in this example rather than giving hypercubes unique names; every hypercube is expressed using one standard hypercube named "Standard [Hypercube]".



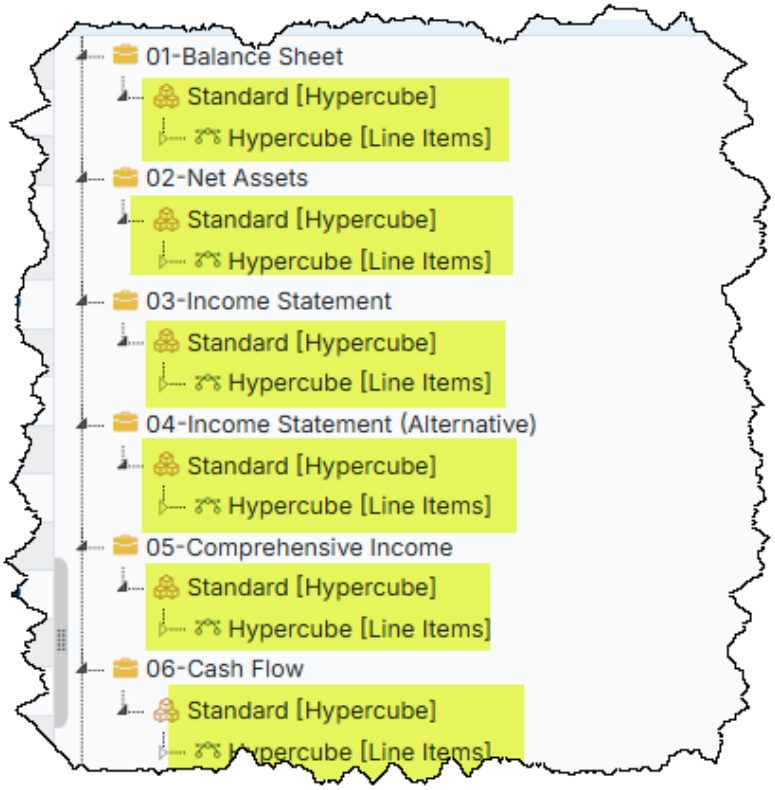
This approach has advantages and consequences. One advantage is that those creating hypercubes do not have to think of a name for the hypercube. But that brings

a disadvantage in the fact that software cannot use the hypercube name as an identifier to, say, extract information from a report. It is still possible to extract information from a report by hypercube; but an additional step is necessary to identify the information you want to extract using prototype theory.

Another advantage is that this basically requires that there be a one-to-one relationship between a network and a hypercube.

2.5. PROOF-Hypercubes3

This is exactly the same as Hypercube2 except that both "Standard [Hypercube]" and "Hypercube [Line Items]" are used.



There is really little difference between PROOF-Hypercubes2 and PROOF-Hypercubes3 except the naming/labeling of the [Line Items] report element of each hypercube. If you give each hypercube the same standard name/label "Standard [Hypercube]" then what does giving the [Line Items] different names/labels get you? Does not seem like much if anything at all.

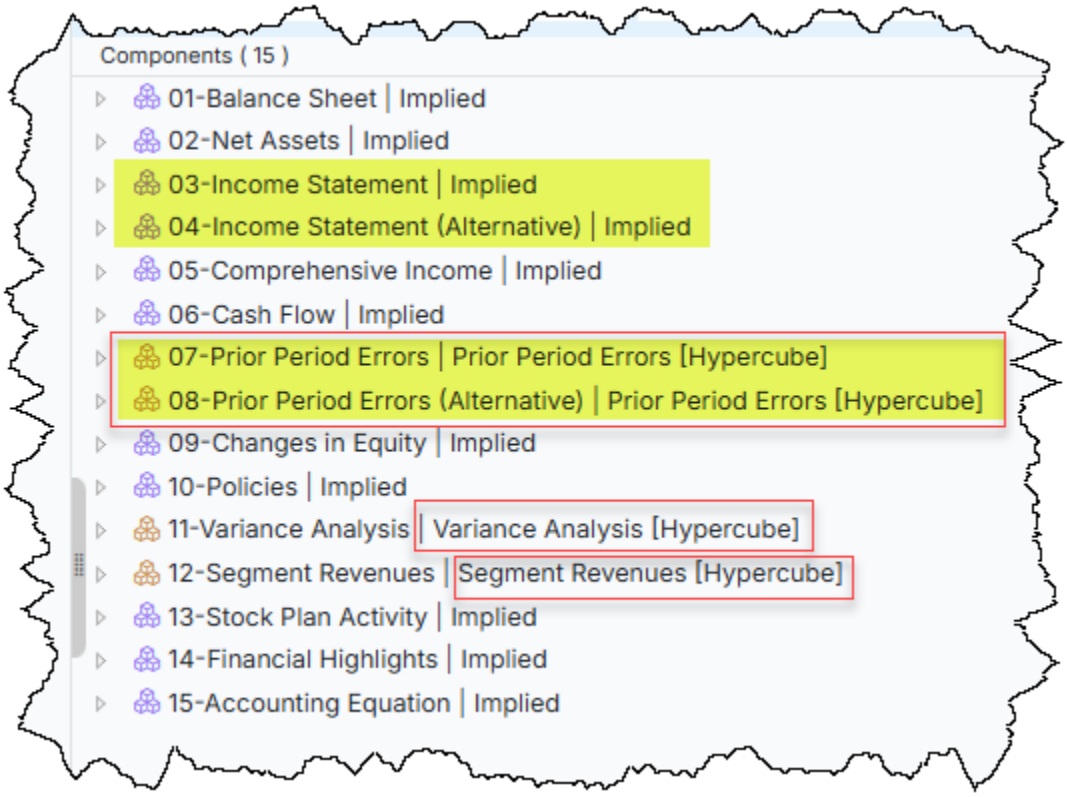
However, like naming/labeling the hypercube buys you not having to come up with those names/labels; the same reasoning can be used to conclude that using a standard identifier for the [Line Items] such as "Hypercube [Line Items]" saves work when creating an XBRL taxonomy.

Another consequence is that it is an even firmer requirement that each hypercube be within its own network because it is literally impossible to model individual hypercubes in the same network because there would be conflicts when modeling the details of each different "Hypercube [Line Items]" container.

Using "Hypercube [Line Items]" makes sense to use on uniquely named/labeled hypercubes also such as in PROOF-Master or PROOF-Alternative1 or PROOF-Alternative2. The [Line Items] of a hypercube actually act somewhat as a dimension. Also, a hypercube can only have exactly one set of [Line Items].

2.6. PROOF-Implied

This example is exactly the same as the Master or baseline example except that explicit hypercubes where only used when they are required because noncore dimensions must be used to represent the information contained in a financial disclosure.



Above you see the "Components" view of the structures of a report. Remember that a Component is a network plus a hypercube. You need the notion of a component in order to be able to distinguish hypercubes that use the same name/label as an identifier for the hypercube. For example, note "Prior Period Errors [Hypercube]" being used above in the networks with the numbers 07 and 08. You can only tell the difference between the two because of the network information provided along with the information about the hypercube.

An "implied [Hypercube]" is the idea that every network is itself a hypercube after excluding all other information from the network contained within other hypercubes within that network.

Saying this another way; a network can contain information represented within one or more hypercubes; plus, in can also contain information not represented within any hypercube. All information represented within a network that is not represented within some other explicitly defined hypercube exists in the notion of an "Implied [Hypercube]" that exists virtually (i.e. it does not physically exist).

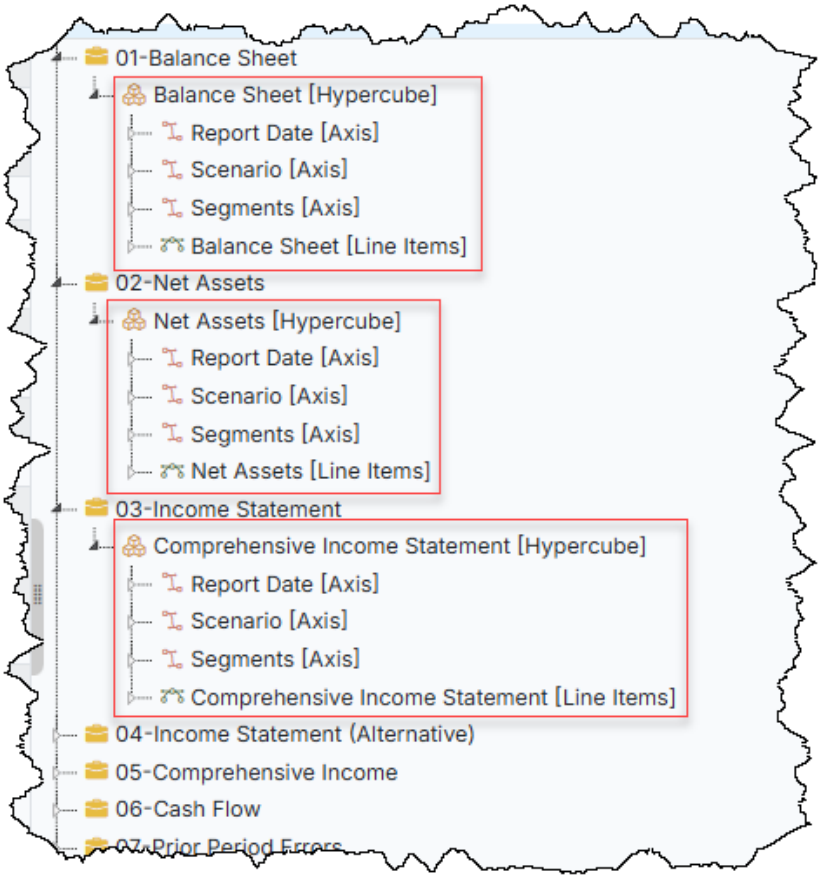
Why can't all hypercubes be implied? This is impossible because in order to add additional noncore dimensions requires that an explicitly defined hypercube be created and then used to represent the additional noncore dimensions within that hypercube.

An "Implied [Hypercube]" can only have core dimensions within that hypercube. Someone creating an XBRL taxonomy can never assign additional noncore dimensions to an "Implied [Hypercube]".

Again, the notion of a "Component" and of "Implied [Hypercube]" do not exist within the XBRL technical specification. They are useful ideas that help one discuss and work with things that are unexplained by XBRL.

2.7. PROOF-Dimensions

This example is exactly the same as the Master or baseline example except that every hypercube is explicitly defined (i.e. there are no implied hypercubes) and in addition every dimension is likewise explicitly defined (i.e. if a dimension was used anywhere, it will be explicitly shown everywhere).



There are two points being made here. First, per the rules of XBRL Dimensions; every dimension default is global in nature and exists on every fact in a report. This is true no matter which hypercube you are looking at and it also is true for facts that exist within no hypercube (i.e. within an implied hypercube).

Second, sometimes those assigning dimensions to hypercubes in a model have the belief that dimensions in a report model assign properties to facts.

Report models should not define properties for facts; base XBRL taxonomies should define properties. Why? If report models were to define properties than different report models could define different properties.

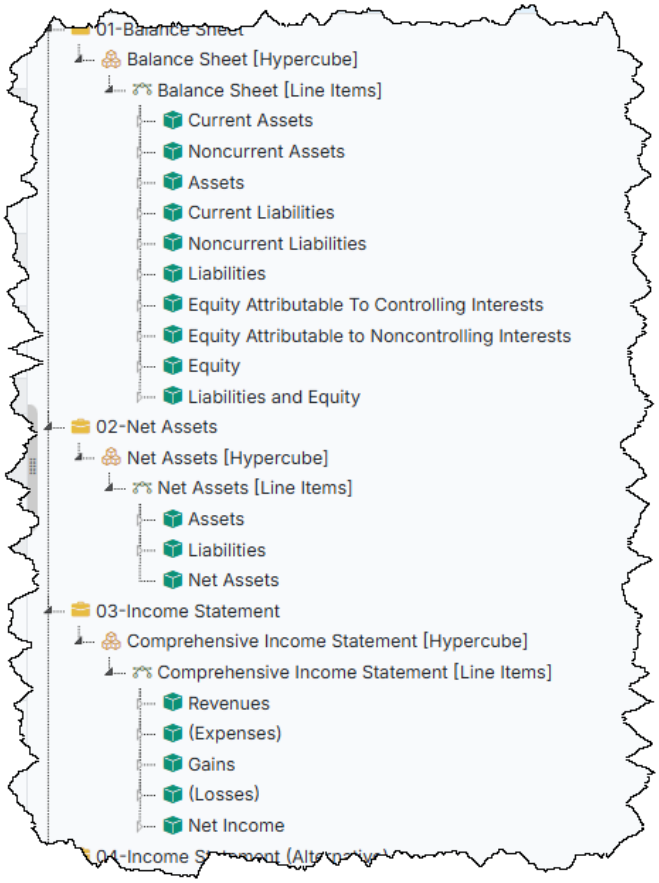
Report models should define properties for extension concepts defined for a report model. Or, report models should "anchor" extension concepts to a base taxonomy and then inherit properties from the anchor point within a base taxonomy.

When to define dimensions for a hypercube within a report model is currently unclear and ambiguous. When dimensions should be provided and exactly what is meant when dimensions are provided should be consistent for every reporting economic entity.

It would be very hard to argue that the PROOF-Dimensions representation and the PROOF-Master and even the PROOF-Implied have different meaning. If there are differences in meaning, then obviously it should be possible to precisely explain those differences.

2.8. PROOF-Blocks

This example is exactly the same as the Master or baseline example except that every possible [Abstract] report element, used to organize the XBRL presentation relations, was removed from the model.



It is easer to see the differences when looking at the human readable rendering generated from the XBRL based information provided within the report model. Here

is the human rendering of the first network and hypercube of the PROOF-Blocks example:

Concept [Aspect]	Period [Aspect]	
	2023-12-31	2022-12-31
Current Assets	500	0
Noncurrent Assets	3000	0
Assets	✓ 3500	✓ 0
Current Liabilities	0	0
Noncurrent Liabilities	0	0
Liabilities	✓ 0	✓ 0
Equity Attributable To Controlling Interests	3000	0
Equity Attributable to Noncontrolling Interests	500	0
Equity	✓ 3500	✓ 0
Liabilities and Equity	✓ 3500	✓ 0

For contrast, here is exactly the same rendering for the PROOF-Master which does contain the abstract report elements which were removed from the PROOF-Blocks representation shown above:

Concept [Aspect]	Period [Aspect]	
	2023-12-31	2022-12-31
Assets [Roll Up]		
Current Assets	500	0
Noncurrent Assets	3000	0
Assets	✓ 3500	✓ 0
Liabilities and Equity [Roll Up]		
Liabilities [Roll Up]		
Current Liabilities	0	0
Noncurrent Liabilities	0	0
Liabilities	✓ 0	✓ 0
Equity [Roll Up]		
Equity Attributable To Controlling Interests	3000	0
Equity Attributable to Noncontrolling Interests	500	0
Equity	✓ 3500	✓ 0
Liabilities and Equity	✓ 3500	✓ 0

Notice the four abstract report elements shown in bold in the second human readable rendering (PROOF-Master) that do not exist in the first human readable rendering (PROOF-Blocks).

Arguably, both human readable representations are very logical and easy to read. If you look at each of the human readable renderings of the PROOF-Blocks which have

the abstract report elements removed, you can see that each of the human readable representations are readable and understandable.

So, what is the purpose of the abstract report elements? Well, I think that one could also agree that the abstract elements in the second human readable rendering do add a bit more clarity to the representation. The abstract report elements provided certainly don't hurt anything. They serve as logical containers that make the information just a little bit easier to read.

The primary point though is that the information conveyed is exactly the same whether the abstract report elements exist or whether they do not exist. The abstract report elements do not impact the meaning of what is reported, only, perhaps, the presentation of what is presented.

2.9. PROOF-Sparse

This example is exactly the same as the Alternative2 except that in this example a hypercube that represents a segment breakdown is intermingled with multiple other blocks of information causing a "sparse" hypercube as a result (i.e. a hypercube with a lot of blank facts because of a non-best practices use of hypercubes).

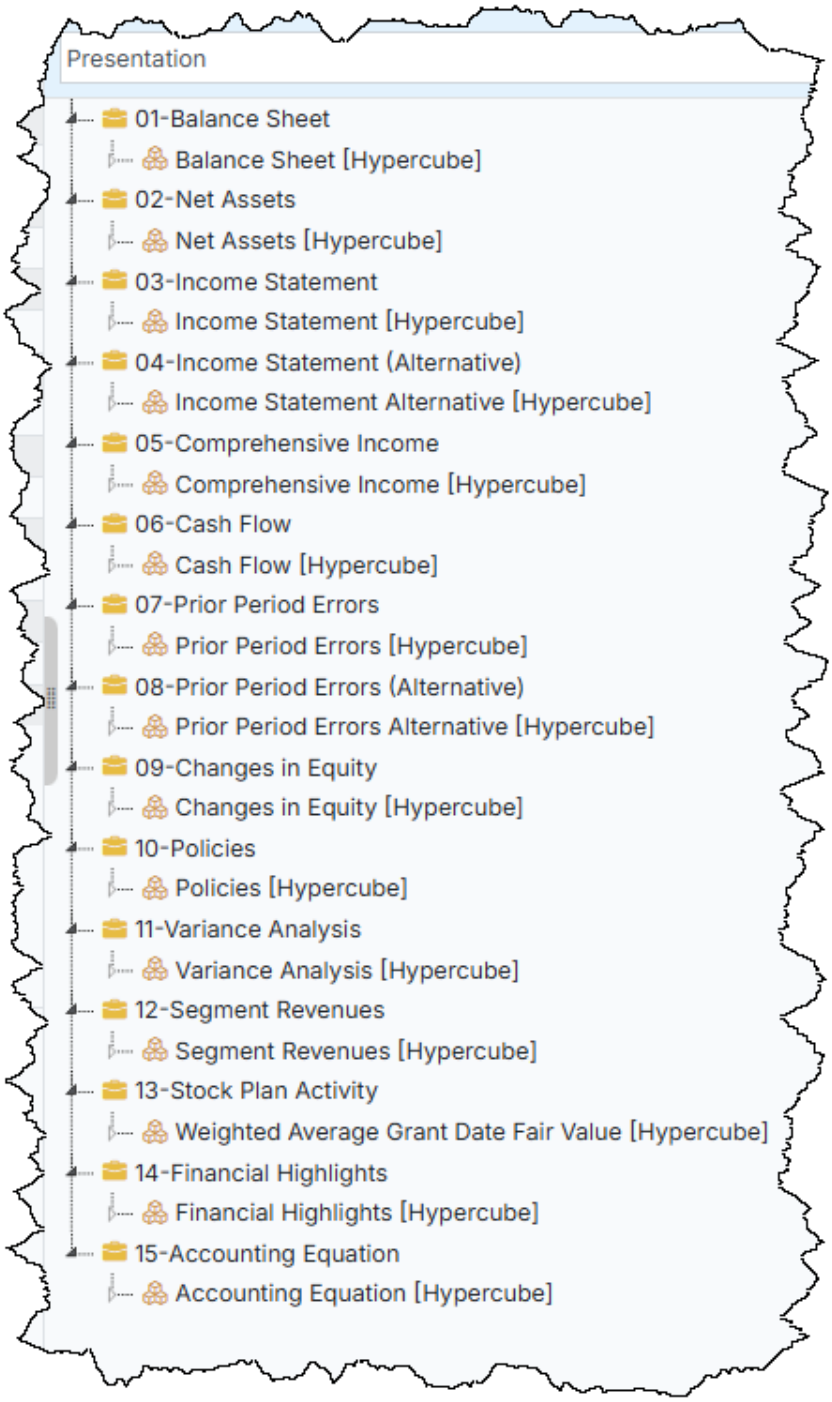
Concept [Aspect]	Units	Period [Aspect]			
		2023-01-01 | 2023-12-31			
		Segments [Axis]			
		Segment Alpha [Member]	Segment Bravo [Member]	Segment Charlie [Member]	All Segments [Member]
Assets [Roll Up]					
Current Assets	iso4217:USD				$ 500
Noncurrent Assets	iso4217:USD				3,000
Assets	iso4217:USD				3,500
Liabilities and Equity [Roll Up]					
Liabilities [Roll Up]					
Current Liabilities	iso4217:USD				$ 0
Noncurrent Liabilities	iso4217:USD				0
Liabilities	iso4217:USD				0
Equity [Roll Up]					
Equity Attributable To Controlling Interests	iso4217:USD				3,000
Equity Attributable to Noncontrolling Interests	iso4217:USD				500
Equity	iso4217:USD				3,500
Liabilities and Equity	iso4217:USD				$ 3,500
Comprehensive Income [Roll Up]					
Revenues	iso4217:USD	$ 1,000	$ 4,000	$ 2,000	$ 7,000
(Expenses)	iso4217:USD				(3,000)
Gains	iso4217:USD				1,000
(Losses)	iso4217:USD				(2,000)
Net Income	iso4217:USD				$ 3,000
Net Assets [Roll Up]					
Assets	iso4217:USD				3,500
Liabilities	iso4217:USD				0
Net Assets	iso4217:USD				$ 3,500
Net Cash Flow [Roll Up]					
Net Cash Flow Operating Activities	iso4217:USD				$ 1,500
Net Cash Flow Investing Activities	iso4217:USD				1,000
Net Cash Flow Financing Activities	iso4217:USD				1,000
Net Cash Flow	iso4217:USD				$ 3,500
Assets [Roll Forward]					
Assets, Beginning Balance	iso4217:USD				$ 0
Net Cash Flow	iso4217:USD				3,500
Assets, Ending Balance	iso4217:USD				$ 3,500
Changes in Equity [Roll Forward]					
Equity, Beginning Balance	iso4217:USD				$ 0
Comprehensive Income	iso4217:USD				3,000
Investments by Owners	iso4217:USD				1,000
(Distributions to Owners)	iso4217:USD				(500)
Equity, Ending Balance	iso4217:USD				$ 3,500
Weighted Average Grant Date Fair Value [Roll Forward Info]					
Nonvested Fair Value, Beginning Balance	iso4217:USD/shares				32.72
Granted	iso4217:USD/shares				41.51
Vested	iso4217:USD/shares				30.92
Forfeited	iso4217:USD/shares				35.93
Nonvested Fair Value, Ending Balance	iso4217:USD/shares				36.92
Financial Highlights [Set]					
Revenues	iso4217:USD	$ 1,000	$ 4,000	$ 2,000	$ 7,000
Comprehensive Income	iso4217:USD				3,000
Distributions to Owners	iso4217:USD				500
Segment Revenues [Set]					
Revenues	iso4217:USD	$ 1,000	$ 4,000	$ 2,000	$ 7,000
Accounting Equation [Arithmetic]					
Assets	iso4217:USD				3,500
Liabilities	iso4217:USD				0
Equity	iso4217:USD				3,500
Basis of Reporting [Text Block]					This is the basis of reporting.
Nature of Operations [Text Block]					This is the nature of operations.
Revenue Recognition Policy [Text Block]					This is the revenue recognition policy.

The point of this example is to show the consequences of not letting the information itself drive the representation of the information. While the representation can be considered logical, it tends to be confusing to read and make sense of.

It also can make creating software more challenging because the software needs to consider and properly process many different possible permutations and combinations.

2.10. PROOF-Proper

This example is very similar to the Master or baseline example except that duplicate hypercubes where made unique and other issues were fixed in this example. This example makes working with the information in the report the easiest it seems.



Notice how there are 15 hypercubes, just like the PROOF-Master, but in this examle every hypercube has a unique name. This enables each block of information and

disclosure to be distinguished from every other block of information by either humans or by machine-based processes. Every hypercube is effectively a uniquely named/labeled object.

3. Consequences of Specific Decisions

The information blocks of each of the ten representations of information within XBRL-based reports is exactly the same. However, working with the information is different depending on choices make. The thing to understand are the consequences of choices made. This can lead to better choices which better align to the goals and objectives one is trying to achieve.

3.1. Hypercube Use and Naming

There are three different approaches to the use and naming of hypercubes in XBRL-based reports. One alternative, not using hypercubes at all, is impossible because it is obvious that the logic used within financial reports demands the capabilities offered by XBRL Dimensions and the hypercubes that specification brings to the table. Noncore dimensions are necessary to effectively represent financial information within XBRL. As such, not using hypercubes at all is not an alternative.

XBRL International has published guidance, *Technical Considerations for the use of XBRL Dimensions 1.0*[13]*,* that suggests against mixing dimensional and nondimensional models. Further, it is impossible to query a report for a machine readable token if there is no way to represent that machine readable token. Meaning, if a hypercube does not exist; then you cannot query information using that hypercube name.

On the other hand, it is a lot of work creating and naming/labeling hypercubes.

3.2. Supplementing using Prototype Theory

Even if hypercube names do not explicitly exist, it is still very possible to extract information using externally defined information.

There are two ways to identify something. The first is described in the section above, using an identifier or token to identify a set of information, for example a hypercube.

The second way to identify something is to examine the parts you see and use the parts to identify what you are looking at.

So, if a hypercube or other identifier does not exist; a set of rules can be defined to examine parts of something and then an identifier can be created to describe what is being identified.

This is how the disclosures and disclosure mechanics rules of the Seattle Method work.

3.3. Disclosures

Ultimately, what is of interest in an XBRL-based report is not networks or hypercubes or blocks of information; rather it is the disclosures represented within the networks and hypercubes represented as information blocks.

---

[13] XBRL International, *Technical Considerations for the use of XBRL Dimensions 1.0*, https://www.xbrl.org/WGN/dimensions-use/WGN-2015-03-25/dimensions-use-WGN-2015-03-25.html

There are two paths to getting to that disclosure information. The first path is using explicit identifiers to name each disclosure; the second path is to look at the information that was disclosed and figuring out what disclosure that information is providing.

In each of the 10 cases, disclosures can be identified whether or not hypercubes have been provided, it makes on difference which network the information is represented in, abstract concepts make no difference in terms of identifying the information.

1. **Master**: https://auditchain.infura-ipfs.io/ipfs/QmRvuNK8JLuTXAhCEik6uDF7N6NCiL4HvdmTjXTgvxbmYm/blocks.html
2. **Alternative1**: https://auditchain.infura-ipfs.io/ipfs/QmYVkXhnZuwstkjXjdyHCL43RacSsLVugzFurjdM9BNUK1/blocks.html
3. **Alternative2**: https://auditchain.infura-ipfs.io/ipfs/QmXdCJTL7GUzbPWNptZnN4WAbwhGFSwKoGEnePbg3Qq1By/blocks.html
4. **Hypercubes2**: https://auditchain.infura-ipfs.io/ipfs/QmeKuk7JJMpjhJXWr6KdAADruUhB6JrjrkjagRak3uWUq8/blocks.html
5. **Hypercubes3**: https://auditchain.infura-ipfs.io/ipfs/QmXVQQ9YHPkh3EnzovzbNcAFyvp36s5ivG1sKn2mG4SxCj/blocks.html
6. **Implied**: https://auditchain.infura-ipfs.io/ipfs/QmTzuW6gRGXsZjgHencMtRxzQfDY1E9NkgDQYQBRGfMj1M/blocks.html
7. **Dimensions**: https://auditchain.infura-ipfs.io/ipfs/QmeA7DeYUDrg24L2eMYGjJfKbbEJBiFMQTAaeHd9tbwHLg/blocks.html (This is a software bug related to the extra dimensions)
8. **Blocks**: https://auditchain.infura-ipfs.io/ipfs/QmQWcoiYAd4SM9FYaGrHPTSt4yaawHf4NZ7wx2ckR8EB29/blocks.html
9. **Sparse**: https://pacioli.auditchain.finance/reportAnalysis/f993e56f12bca1cd9d10330185b28cf72f148ae6.report/disclosures.html (This is a software bug related to discover of the changes in equity roll forward)
10. **Proper**: (need to create rules, report is local) https://auditchain.infura-ipfs.io/ipfs/QmYdVvfYb4NcTiXgUbccTV6us7169i3AcC9QZJ4GW2jrTF/blocks.html