Essentials of XBRL-based Digital Financial Reporting (Platinum)

Understanding and using modern approaches to financial reporting when the report model can be modified by report creators

by

Charles Hoffman, CPA

A resource for professional accountants who want to understand and otherwise work with XBRL-based digital financial reports including external financial reporting managers, internal auditors, external auditors, financial analysts, regulators, and other business professionals. A resource for software engineers who want to implement useful software which will be used by professional accountants. Excellent resource for training trainers who will then train others.

Second Edition
DRAFT of 2023-08-13

http://www.xbrlsite.com/seattlemethod/platinum/index.html

About the author:

Charles Hoffman, CPA, is credited as being the *Father of XBRL*. He started his public accounting career as an auditor with the international firm then called Price Waterhouse, served various roles in industry and public accounting for over 25 years, and has worked with XBRL since its introduction by the AICPA in 1998. In 2006, he received the AICPA Special Recognition Award for his pioneering role in developing XBRL. He has authored numerous publications including *XBRL for Dummies*, a number of *Journal of Accountancy* articles, writes a blog relating to XBRL, and contributed to a number of XBRL related technical specification and best practices documents. Currently, Charlie works as a consultant to CPAs and software vendors who want to better understand the subtle details of this new digital medium.

Charlie was co-editor of the first ever US GAAP XBRL taxonomy, contributor to the *XBRL 2.1 Specification* and the *XBRL Dimensions* specification, editor of the *Financial Reporting Taxonomy Architecture* and *Financial Reporting Instance Standards*, co-author of the *US GAAP Taxonomy Architecture*, part of the project team which created the *US GAAP Taxonomy*, and a major contributor to the IFRS XBRL taxonomy for a five-year period, and consultant to numerous other XBRL taxonomy projects.

Acknowledgements:

Copyright (full and complete release of copyright)

All content of this document is placed in the public domain. We hereby waive all claim of copyright in this work. This work may be used, altered or unaltered, in any manner by anyone without attribution or notice to us. To be clear, we are granting full permission to use any content in this work in any way you like. We fully and completely release all rights to any copyright on this content. If you feel like distributing a copy of this work, you may do so without attribution or payment of any kind. All that said, attribution is appreciated should one feel so compelled. The copyrights of other works referenced by this document are established by the referenced work.

"I skate to where the puck is going to be, not where it has been." Wayne Gretzky, legendary Canadian hockey star

"Being ignorant is not so much a shame, as being unwilling to learn." Benjamin Franklin

"*Learn the rules like a pro, so you can break them like an artist*." Pablo Picasso

A **best practice** is a method or technique that has been generally accepted as superior to any other known alternatives because it produces results that are superior to those achieved by other means or because it has become a standard way of doing things.

Best practices (a.k.a. good practices) are techniques that have produced outstanding results in other situations, inside or outside of a particular organization and which can be validated, codified, and shared with others and recommended as models to follow[1].

Proficiency is the capability, skill, and knowledge that you might have for doing something. There are general levels of proficiency: literacy, fluency, mastery. Proficiency is a progression. This document will provide XBRL-based reporting literacy and fluency.

This document helps professional accountants and software engineers understand the essentials of XBRL-based digital financial reports, the dynamics of those reports, and a best practice based (good practice) control mechanism or method[2] that that has proven to consistently yield high-quality XBRL-based financial reports where the report model can be "reshaped" or "altered" or "modified" by report creators. US GAAP, IFRS, and UK GAAP are such financial reporting schemes.

The *Platinum XBRL-based Financial Report Examples*[3] have all been created using this method which I call the *Seattle Method*. Ultimately, XBRL + SBRM could become an enhanced or improved approach to implementing business intelligence (BI)[4]. Another way to look at this is a new modern approach to creating professional spreadsheets[5]. Yet another way is to understand that accounting working papers and audit schedules could be represented using XBRL+SBRM[6].

Fundamentally, a financial report is a knowledge graph[7]. That knowledge graph has a logical schema[8] that enforces the rules of the financial report and financial reporting. Software, such as expert systems for creating[9] and consuming financial report information can make use of this financial report knowledge graph if the information is effectively represented in machine-readable form.

---

[1] European Guide to good Practice in Knowledge Management - Part 5: KM Terminology, page 3 (PDF page 9), http://arielsheen.com/wp-content/uploads/2019/10/CEN-CWA14924-05-2004-Mar.pdf#page=9

[2] *Seattle Method*, http://xbrlsite.com/seattlemethod/SeattleMethod.pdf

[3] Platinum XBRL-based Financial Report Examples, https://digitalfinancialreporting.blogspot.com/2023/07/platinum-xbrl-based-financial-report.html

[4] XBRL+SBRM=Improved BI, https://digitalfinancialreporting.blogspot.com/2023/08/xbrlsbrm-improved-bi.html

[5] *Modern Spreadsheet*, https://digitalfinancialreporting.blogspot.com/2023/05/modern-spreadsheet.html

[6] *Creating New Tool for Accounting and Audit*, https://digitalfinancialreporting.blogspot.com/2023/08/creating-new-tool-for-accounting-and.html

[7] Charles Hoffman, CPA, *Financial Report Knowledge Graphs*, http://xbrlsite.azurewebsites.net/2021/Library/FinancialReportKnowledgeGraphs.pdf

[8] Charles Hoffman, CPA, *Logical Schema of Financial Reports*, http://xbrlsite.com/seattlemethod/LogicalSchemaOfFinancialReports.pdf

[9] World's First Expert System for Creating Financial Reports, https://www.youtube.com/playlist?list=PL80qjzvfqwtNuTekdlRy0rhaHEDIXkOh3

1. Introduction

In this section we will take a very basic introductory look at the physical and logical details of financial reports, the XBRL-based approach to creating those reports and report models, and provide an over-arching framework that is helpful in pulling the details involved together. The goal is to provide you with a high-level understanding of XBRL-based digital financial reporting and a path to obtaining even greater understanding.

We will expose XBRL to you in layers with each level building on the previous. We are not going to teach you specific software applications, you are on your own for finding the software tools you need. We do point you to some free and open-source software that gets the job done.

We will not be covering the subtleties and nuances of XBRL-based reporting here. We do provide a foundation upon which to build. You do not need to understand accounting or financial reporting to understand this section; but because many people understand the basics of a financial report and because this is targeted mainly at professional accountants and because I am a professional accountant; I will be providing a lot of financial reporting related examples.

To get more details on what we cover, follow the links that are provided or see the document *Mastering XBRL-based Digital Financial Reporting*[10].

1.1. Craft of Creating Financial Statements

Financial statements tell a story. It is a milestone in an accountant's career when the accountant can weave together a logical, cogent, true and fair narrative from information provided and tell a story about an economic entity in the form of a financial statement.

With the ability to perform that task, an accountant has achieved accounting literacy.

That is precisely what an accountant creating an XBRL-based report must also do. But instead of doing this with Microsoft Word, Excel and typing a bunch of stuff into applications that don't understand accounting; with XBRL-based structured information, accountants need to learn a new approach to telling a story about an economic entity. They need to represent that report in terms that both humans and machines, such as software applications, must be able to understand.

Authors Dean Allemang and James Hendler say it well in their book *Semantic Web for the Working Ontologist*[11]:

> "In the hands of someone with no knowledge, they can produce clumsy, ugly, barely functional output; but in the hands of a skilled craftsmen, they can produce works of utility, beauty, and durability."

A brick wall is made of exactly two things: bricks, mortar.

But a brick wall created by a master craftsman, or mason, and a brick wall created by a "weekend warrior" with no knowledge of masonry will be very different. Master craftsmen are created and that process takes time and effort. Skills and experience

---

[10] *Mastering XBRL-based Digital Financial Reporting*, http://xbrl.squarespace.com/mastering-xbrl/

[11] Dean Allemang and James Hendler, *Semantic Web for the Working Ontologist*, page 1, https://www.amazon.com/Semantic-Web-Working-Ontologist-Effective/dp/0123859654

that turn you into a master craftsman need to be accumulated and added to your base of understanding. Accumulating these skills take time, effort, and a great deal of practice.

The term "mason" can be further broken down in to more detailed distinctions. In the Middle Ages, the terms rough masons, row masons, stone setters, layers and freemasons were used to signify differences in skills[12]. Freemasons[13] who were the most skilled were paid the best.

Masons built our physical world, our industrial economy. But now the world is transitioning from a physical industrial-based economy to an information-based economy[14]. Knowledge engineers working with subject matter experts that have the right skills and experience will construct our digital world.

Financial reports are knowledge graphs[15] that are rich with associations. Representing that rich set of associations in the form of machine-readable information creates the capabilities for software applications that can read, understand those associations, and work with that information to perform what seems like magic[16].

But this is not really "magic". It just seems like magic.

Information wants to be free from imperfections. Good tools in the hands of master craftsmen using good methods[17] can produce information that is understandable, useful, durable, and perhaps even beautiful and elegant which truly sets information free!

Even smart software applications appear "dumb" if they are not provided the right information. "Dumb" includes information that is wrong, contradictory, inconsistent, disconnected, and is not otherwise synchronized. On the other hand; properly organizing information enables sophisticated software applications to perform to their potential.

Financial report information is provided in the form of machine-readable models and other metadata that help accountants perform their work effectively.

How exactly to create those associations and what the associations might look like are up to debate. Regardless of how that debate might turn out; associations are important.

Below is a screen show that provides an example of a set of associations from a rather small financial reporting scheme[18]:

---

[12] How are Masons and Freemasons Different?, https://www.jeffersoncountylodge.org/post/2019/08/11/how-are-masons-and-freemasons-different

[13] Wikipedia, *Freemasons*, https://en.wikipedia.org/wiki/Freemasonry

[14] Charles Hoffman, CPA, The Great Transmutation, http://xbrlsite.azurewebsites.net/2022/Library/TheGreatTransmutation.pdf

[15] Charles Hoffman, CPA, *Financial Report Knowledge Graphs*, http://xbrlsite.azurewebsites.net/2021/Library/FinancialReportKnowledgeGraphs.pdf

[16] Charles Hoffman, CPA, Expert Systems for Creating Financial Reports Explained in Simple Terms, http://xbrlsite.azurewebsites.net/2022/Library/ExpertSystemForCreatingFinancialReports.pdf

[17] Charles Hoffman, CPA, Seattle Method, http://xbrlsite.com/seattlemethod/

[18] Associations in a small financial reporting scheme, http://accounting.auditchain.finance/reporting-scheme/mini/reference-implementation/pacioli/typeSubTypeGraph.html
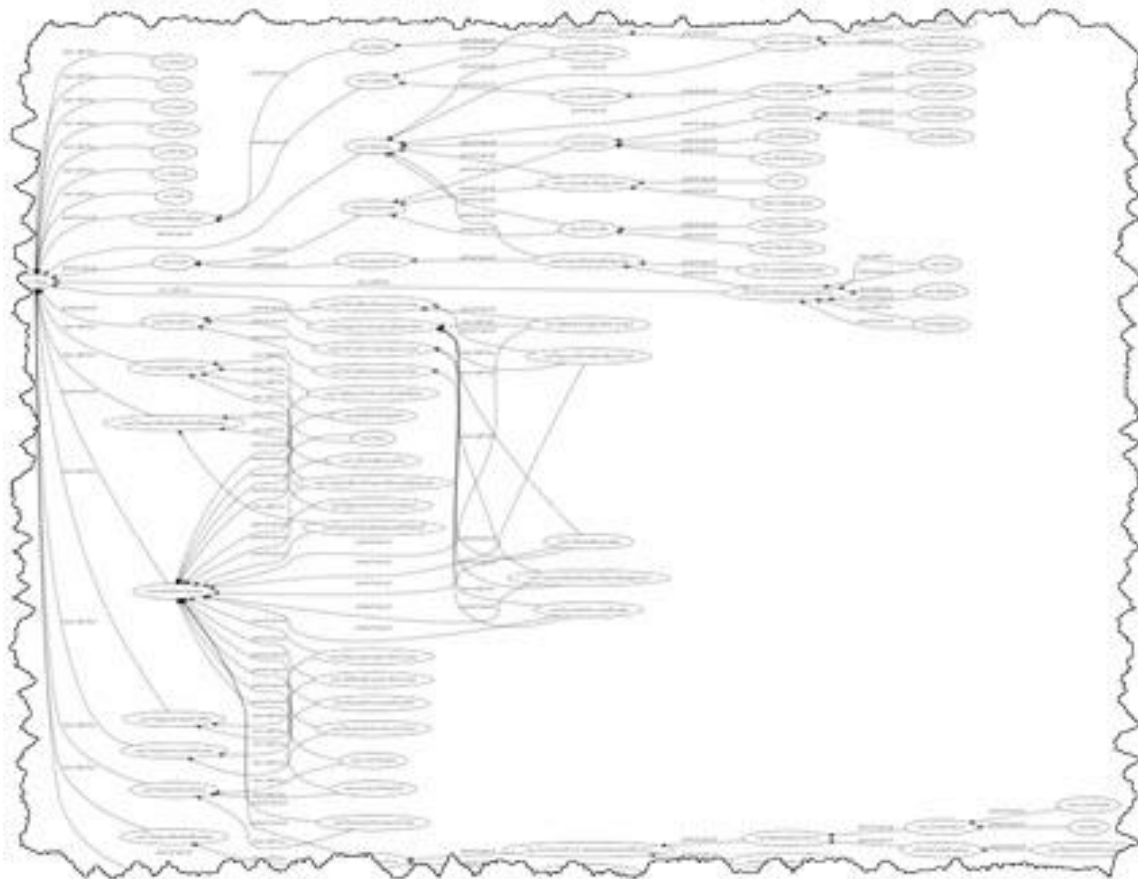
The example of associations shown above and the many other examples[19] that I provide help you understand both the possibilities and the many details that are involved. To an untrained observer these associations mean less. But to a trained, skilled master craftsman these samples and examples look very different.

XBRL-based digital financial reporting is a tool that needs to be learned.

David Weinberger points out in his book *Everything Is Miscellaneous*[20] two very important things:

- That every classification scheme ever devised inherently reflects the biases of those that constructed the classification system.

- The role metadata plays in allowing you to create your own custom classification system so you can have the view of something that you want.

Mr. Weinberger continues in his book by pointing out the notion of the three orders of order:

- **First order of order**. Putting books on shelves is an example the first order of order.

---

[19] PLATINUM Best Practices, Examples, Conformance Suite, https://digitalfinancialreporting.blogspot.com/2023/07/platinum-business-use-cases-test-cases.html

[20] David Weinberger, Everything is Miscellaneous, https://www.amazon.com/Everything-Miscellaneous-Power-Digital-Disorder/dp/0805088113

- **Second order of order**. Creating a list of books on the shelves you have is an example of second order of order. This can be done on paper or it can be done in a database.

- **Third order of order**. Adding even more information to information is an example of third order of order. Using the book example, classifying books by genre, best sellers, featured books, bargain books, books which one of your friends has read; basically, there are countless ways to organize something.

Associations[21] and classification[22] are two very important tools of the information age. Those that understand how to use these tools and techniques and create this critically important metadata[23], those who are craftsmen and perhaps the information age equivalent to a freemason will construct our digital world.

XBRL is another medium[24] for telling the story, but the story remains the same whether it is conveyed with structured information like XBRL or with some other format such as PDF, HTML, Excel, or Word.

A truly financial literate accountant can tell the story correctly and consistently no matter what medium is used to convey the information as long as they are skilled in using the tools of the profession to convey the information in the required format. The external financial report is only one slice of the story; internal reports, budgets, cash flow forecasts, and other such reports are other slices of the same story.

Information must be considered in context to be properly understood. The bigger picture must also be considered. Accountants balance all these moving parts to tell the story of an economic entity.

While not every accountant needs to be a master craftsman in the art and science of XBRL-based digital financial report creation; those helping software engineers create software and those creating important base taxonomies used by many others do need to be master craftsmen.

1.2. *Essence of a General-Purpose Financial Report*

A general-purpose financial report is a high-fidelity, high-resolution, high-quality information exchange mechanism. The report is a collection of values and disclosures (complex logical information) required by statutory and regulatory requirements plus whatever management of an economic entity wants to voluntarily disclose. The report represents quantitative and qualitative information about the financial condition and financial performance of an economic entity. That financial report is expected to be a true and fair representation of information about the reporting economic entity. There are a number of different financial reporting schemes[25] that might be used to create that report: US GAAP, IFRS, IPSAS, GAS, FAS, etc.

---

[21] Random Blather, *Information Isn't Power*, https://random-blather.com/2014/04/28/information-isnt-power/

[22] Classification, http://xbrl.squarespace.com/journal/2019/5/14/understanding-the-power-of-classification.html

[23] *Central Role Played by Ontology-like Things and Other Metadata*, http://xbrl.squarespace.com/journal/2019/7/30/central-role-played-by-ontology-like-things-and-other-metada.html

[24] Understanding XBRL is a Knowledge Media, http://xbrl.squarespace.com/journal/2017/1/16/understanding-that-xbrl-is-a-knowledge-media.html

[25] Charles Hoffman, CPA, *Comparison of Financial Reporting Schemes High Level Concepts*, http://xbrlsite.azurewebsites.net/2018/Library/ReportingSchemes-2018-12-30.pdf

Financial reports are not forms. Financial reports are not uniform. Financial reports have variability allowed by reporting rules[26]. This consciously allowed variability is an essential, characteristic trait of robust reporting schemes such as US GAAP, IFRS, and others. This variability contributes to the richness, high-fidelity, and high-resolution of reported financial information that is unique to an industry sector, a style of reporting, or an economic entity. This variability is a feature of such reporting schemes. Different reporting styles, different subtotals used to aggregate details, and using some specific approach given a set of allowed alternatives are examples of variability. Variability does not mean "arbitrary" or "random". There are known identifiable patterns.

And so, any technical implementation of financial reporting needs to be able to properly, correctly, and fully support that allowed variability so that those reporting information and those consuming that reporting information are on the same page. Report creators can adjust, reshape, and otherwise modify a financial report model within permitted bounds. Those modifications must be controlled appropriately in order to achieve the necessary quality and an effective information exchange. And keep in mind that one of the users of this information could be a machine such as an intelligent software agent.

Saying this another way, the semantic hygiene[27] of an XBRL-based digital financial report must be consciously understood and maintained by professional accountants.

Consider the following use case of a general-purpose financial report:

> Two economic entities, A and B, each have information about their financial position and financial performance. They must communicate their information to an investor who is making investment decisions which will make use of the combined information so as to draw some conclusions. All three parties (economic entity A, economic entity B, investor) are using a common set of basic logical principles (facts, statements, deductive reasoning, inductive reasoning, etc.), common financial reporting standard concepts and relations (i.e. US GAAP, UK GAAP, IFRS, IPSAS, etc.), and a common world view so they should be able to communicate this information fully, so that any inferences which, say, the investor draws from economic entity A's information should also be derivable by economic entity A itself using basic logical principles, common financial reporting standards (concepts and relations), and common world view; and vice versa; and similarly for the investor and economic entity B.

The following is a set of principles related to a general-purpose financial report that would be hard to dispute:

- A general-purpose financial report is a high-fidelity, high-resolution, high-quality information exchange mechanism.

- Prudence dictates that using information from a financial report should not be a guessing game.

- All formats conveying information should convey the exact same meaning be that format paper, e-paper, or some machine-readable format.

---

[26] *Understanding Articulation*, https://digitalfinancialreporting.blogspot.com/2023/08/understanding-articulation.html

[27] *Semantic Hygiene in Financial Reporting*, https://digitalfinancialreporting.blogspot.com/2023/08/semantic-hygiene-in-financial-reporting.html

- Explicitly stated information from information bearers or reliably derived information is preferable to requiring information receivers to make assumptions.

- Double entry bookkeeping enables processes that allow for the detection of information errors and to distinguish errors (unintentional) from fraud (intentional).

- Catastrophic logical failures are to be avoided at all cost as they cause systems to completely fail.

Depicted graphically; the essence of what is taking place when an economic entity, an information bearer, provides information to some information receiver such as an investor, regulator, or lender; is such:
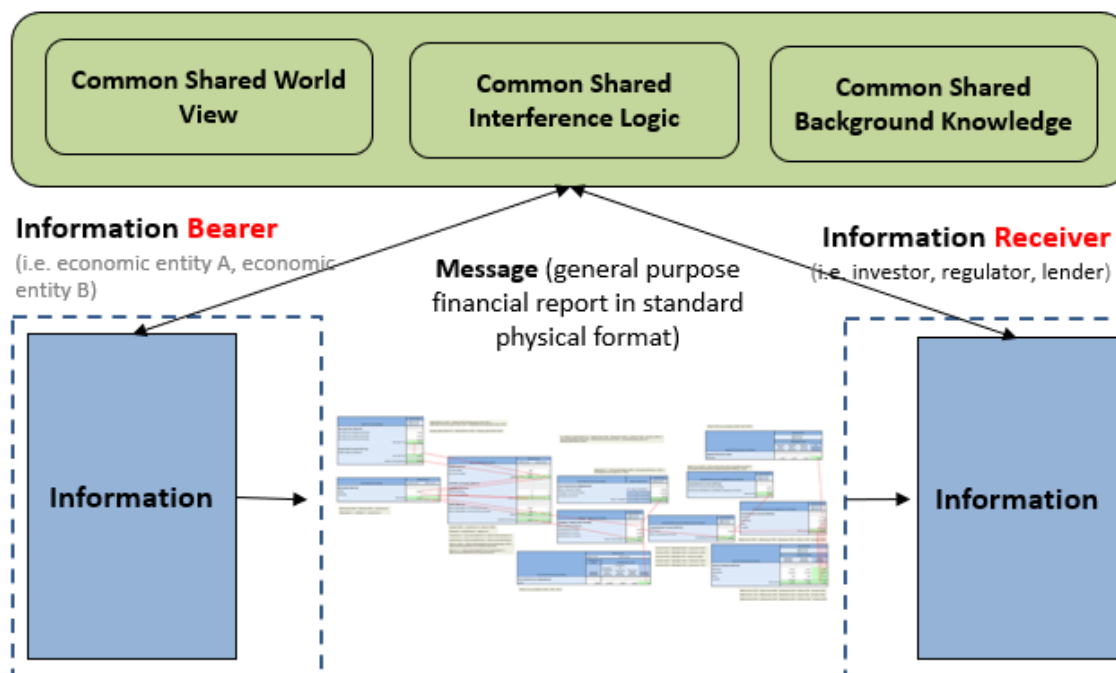
All of this can be described logically in a manner that is easy for a professional accountant to understand.

Rules are used to articulate allowed variability and "channel" creators of reports in the right direction and therefore control variability, keeping the variability within permitted limits. That keeps report quality where it needs to be.

Rules enable things like preventing a user from using a concept meant to represent one thing from unintentionally being used to represent something different. Further, the discipline of describing something in a form a computer algorithm can understand also assists you in understanding the world better; weeding out flaws in your understanding, myths, and misconceptions about accounting and reporting standards.

Given the sophisticated requirements of something like a general-purpose financial report, the XBRL technical syntax needs to be able to reliably deliver specific functionality to make XBRL-based digital general purpose financial reports work appropriately.

1.3. Controlled Customization

As previously pointed out and as explained in more detail in the *Essence of Accounting*[28]; financial statements are not forms.

As such, economic entities creating reports are permitted to make modifications to their report model within specific boundaries. These permitted modifications must be controlled to make sure report creators stay within those permitted boundaries.
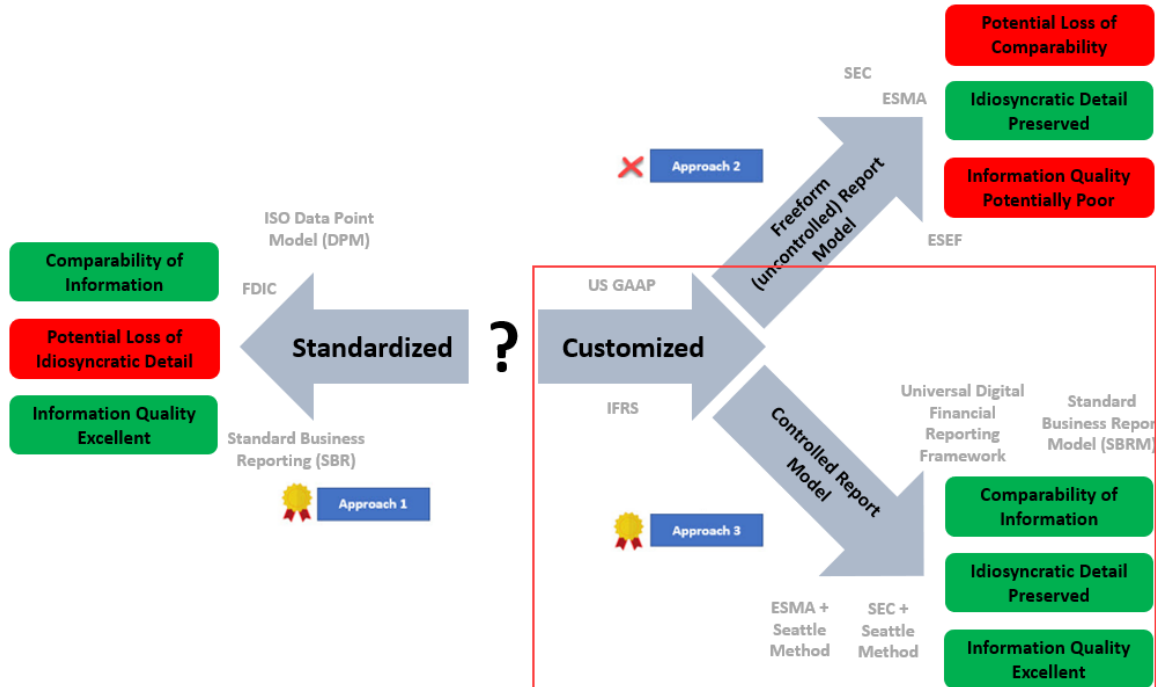
Being ignorant of such boundaries is not a good reason for not staying within those required boundaries.

Given that control of report customizations to keep those customizations within permitted boundaries; that raises the question of how exactly to provide that control.

1.4. How XBRL Works

For step one in your journey into XBRL-based digital financial reporting, I would suggest that you watch the popular video, *How XBRL Works*[29]. The video will supply you with a very brief explanation as to what XBRL actually is and how it works.

If you would like some supplemental background information, also watch the video *XBRL Overview*[30].

In a future and optional section, we are going to show you the ugly, technical details of XBRL. The good news is that you will never need to look at XBRL at this technical level ever again if you do not want to. That level is intended for consumption by a

---

[28] Charles Hoffman, CPA, Essence of Accounting,
http://xbrlsite.azurewebsites.net/2020/Library/EssenceOfAccounting.pdf

[29] YouTube.com, Charles Hoffman, CPA, *How XBRL Works*,
https://www.youtube.com/watch?v=nATJBPOiTxM

[30] YouTube.com, Charles Hoffman, CPA, *XBRL Overview*, https://youtu.be/ibfQwRTP4_c

computer. We are going to focus on the logical description of an XBRL-based financial report. After all, a financial report is a logical system.

1.5. Financial Report is a Logical System

A financial report is a logical system devised by man between about 5,000 and 10,000 years ago[31]. There is nothing natural about financial reports, they do not exist in nature.

A financial report is not an economic entity, it simply conveys certain specific details about the financial status and financial performance of some reporting economic entity. To discuss financial report logical systems, we need a bit of precise jargon which we will provide now in as simple terms as possible to provide that description.

Representation theory[32] is a way of taking complicated objects and "representing" them with simpler objects. Essential elements are provided for and unimportant information is ignored. Knowledgeable domain experts in some area of knowledge determine what is essential and what is unimportant.

Financial reports can be described using a theory such as the *Logical Theory Describing Financial Report*[33]. A financial report must conform to the rules of accounting described by the *Essence of Accounting*[34].

An XBRL-based digital financial report is a machine-readable knowledge graph as explained in *Financial Report Knowledge Graphs*[35]. The financial report knowledge graph must conform to specific logical rules; these rules can be represented and controlled using good practice based mechanisms such as the *Seattle Method*[36]. That way the machine-readable reports can be controlled when they are created making them properly functioning logical systems.

For more detailed information about how computers work please refer to *Computational Professional Services*[37].

So, what exactly is meant by a logical system? Let me provide a simple explanation to help you understand that idea.

1.6. Simple Explanation of Logical System

A **logical system** can be explained by a **logical theory**. A logical theory is an abstract conceptualization[38] of specific important details of some area of knowledge. The logical theory provides a way of thinking about an area of knowledge by means of deductive reasoning to derive logical consequences of the logical theory.

---

[31] Charles Hoffman, CPA, *Financial Report Knowledge Graph*, http://xbrlsite.azurewebsites.net/2021/Library/FinancialReportKnowledgeGraphs.pdf

[32] Representation Theory, http://xbrl.squarespace.com/journal/2021/1/11/representation-theory.html

[33] *Logical Theory Describing Financial Report*, http://xbrlsite.com/seattlemethod/LogicalTheoryDescribingFinancialReport_Terse.pdf

[34] *Essence of Accounting*, http://xbrlsite.azurewebsites.net/2020/Library/EssenceOfAccounting.pdf

[35] *Financial Report Knowledge Graphs*, http://xbrlsite.azurewebsites.net/2021/Library/FinancialReportKnowledgeGraphs.pdf

[36] *Seattle Method*, http://xbrlsite.com/seattlemethod/SeattleMethod.pdf

[37] *Computational Professional Services*, http://xbrlsite.azurewebsites.net/2020/library/ComputationalProfessionalServices.pdf

[38] Wikipedia, *Conceptual Model*, https://en.wikipedia.org/wiki/Conceptual_model

A logical conceptualization provided by a logical theory enables a community of stakeholders trying to achieve a specific goal or objective or a range of goals/objectives to agree on important logical details used for capturing meaning or representing a shared understanding of, and important knowledge in, some area of knowledge or "domain of knowledge".

A logical theory forms a **logical conceptualization** and is made up of a set of logical models, structures, terms, associations, rules, and facts. In very simple terms,

- **Logical conceptualization**: A logical conceptualization is a set of logical models that are consistent with and permissible per the logical theory that describes the logical conceptualization.

- **Model**: A *model*[39] is a set of logical structures that are consistent with and permissible interpretations of that model.

- **Structure**: A *structure* is a set of logical statements which describe the structure.

- **Logical statement**: A *logical statement* is a proposition, claim, assertion, belief, idea, or fact about or related to the area of knowledge to which the logical theory relates. There are five broad categories of logical statements:

    - **Terms**: *Terms* are logical statements that define ideas used by the logical theory such as "assets", "liabilities", "equity", and "balance sheet".

    - **Associations**: *Associations* are logical statements that describe permissible interrelationships between the terms such as "assets is part-of the balance sheet" or "operating expenses is a type-of expense" or "assets = liabilities + equity" or "an asset is a 'debit' and is 'as of' a specific point in time and is always a monetary numeric value".

    - **Rules**: *Rules* are logical statements that describe and check the validity of what tend to be represented as IF…THEN…ELSE types of relationships such as "IF the economic entity is a not-for-profit THEN net assets = assets - liabilities; ELSE assets = liabilities + equity".

    - **Facts**: *Facts* are logical statements about the numbers and words that are provided by an economic entity within a financial report. For example, the financial report might state "assets for the consolidated legal entity Microsoft as of June 20, 2017 was $241,086,000,000 expressed in US dollars and rounded to the nearest millions of dollars.

    - **Properties** are logical statements about the important qualities and traits of the models, structures, terms, associations, rules, and facts.

Fundamentally, a logical theory is a set of logical statements used to describe some logical conceptualization. Those logical statements can be represented in human-readable form or they could be expressed in machine-readable form. Once in machine-readable form, those logical statements can be interrogated using software applications. To the extent that this can be performed effectively; software tools can assist professional accountants, financial analysts, and others working with those logical statements. Those machine readable logical statements can check the validity of a report model and report described by those machine readable logical statements.

---

[39] Wikipedia, *Model Theory*, https://en.wikipedia.org/wiki/Model_theory

1.7. Properly Functioning Logical System

Validity is a state where a logical system is consistent with the logical theory used to describe that logical system; in particular the rules, assertions, and other constraints; and show the logical system is consistent and complete.

In her book *An Introduction to Ontology Engineering*[40], C. Maria Keet, PhD, provides discussion about what constitutes a good and perhaps a not-so-good ontology.

An ontology is part of knowledge; a description of the "things" and the "relations between things". The other part of knowledge is rules. Together and ontology plus rules equals knowledge like the knowledge found in a financial report knowledge graph. There are three specific categories of errors Keet discusses:

- **Syntax errors**: She discusses the notion that a syntax error in an ontology is similar to computer code not being able to compile. For example, when an XBRL processor tells you that the physical syntax of your XBRL taxonomy (report model) or XBRL instance (report) is not valid per the XBRL technical specification.

- **Logic errors**: She discusses the notion of logical errors within information which cause the information to not work as expected. For example, if you represented something in your XBRL taxonomy as a credit when it should have been a debit.

- **Precision and coverage errors**: Finally, Keet discusses the notions of precision and coverage when it comes to judging whether information is good or bad.

Keet uses those terms, others use different terms including sound, complete, and effective to describe a well-functioning logical system. There are other descriptions as well. The following is the description that I will use.

All of these ideas also apply to knowledge graphs.

A logical system is said to be **consistent** with a logical theory if there are no contradictions with respect to the logical statements made by the logical theory that describes the logical system.

A logical theory can have high to low **precision** and high to low **coverage** with respect to describing a logical system.

Precision is a measure of how precisely the information within a logical theory has been represented as contrast to reality of the logical system for the area of knowledge. *Coverage* is a measure of how completely information in a logical theory has been represented relative to the reality of the logical system for the area of knowledge.

When a logical system is consistent with some logical conceptualization provided by a logical theory and that logical system has high precision and high coverage the logical system can be considered a **properly functioning logical system**.

---

[40] C. Maria Keet, PhD*, An Introduction to Ontology Engineering*, PDF page 23, https://people.cs.uct.ac.za/~mkeet/files/OEbook.pdf#page=23

When a logical system is properly functioning, that logical system creates what can be described as a virtuous cycle[41].

Define terms, associations, structures, rules, models

Extract facts report using terms, associations, structures, rules, model

Create facts reported using terms, associations, structures, rules, model

Virtuous Cycle

Verify facts conveyed in report consistent with terms, associations, structures, rules, model
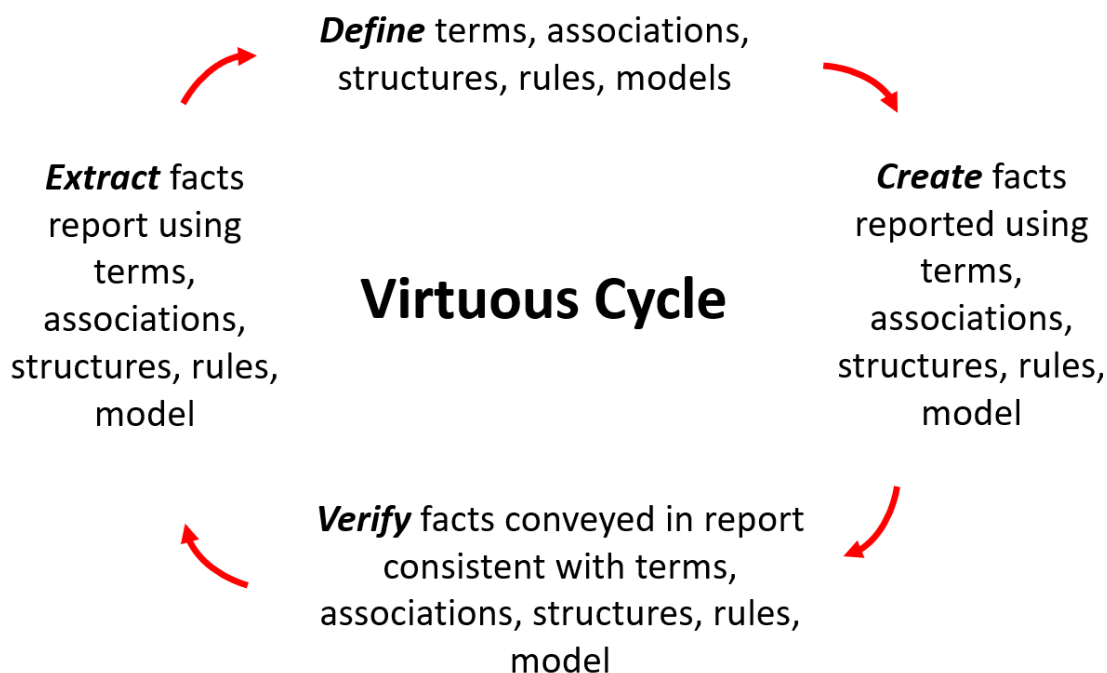
A logical theory conveys knowledge about a logical conceptualization and that knowledge, in the form of an ontology plus rules, can be represented within a machine-readable knowledge graph system[42]. With such a knowledge graph system one can describe a financial report knowledge graph system[43].

Such a knowledge graph system for financial reporting enables the creation of new types of tools for financial accounting, reporting, auditing, and analysis[44]. These new tools, such as the modern semantic spreadsheet[45], will change the universal technology of accountability[46]. Software engineers can help augment the skills of an accountant much like a calculator enhances an accountant's ability to do math.

For more detailed information related to logical theories and logical systems, please see *Logical Systems*[47].

Let us now walk through a very simple but tangible example to help you better understand the moving pieces of the puzzle. While this simple example might seem

---

[41] Charles Hoffman, CPA, *Virtuous Cycle*, http://xbrl.squarespace.com/journal/2020/4/29/virtuous-cycle.html

[42] Knowledge Graph System, https://digitalfinancialreporting.blogspot.com/2023/07/knowledge-graph-system.html

[43] *Knowledge Graph System for Financial Reporting*, https://digitalfinancialreporting.blogspot.com/2023/07/knowledge-graph-system-for-financial.html

[44] Creating New Tool for Accounting and Audit, https://digitalfinancialreporting.blogspot.com/2023/08/creating-new-tool-for-accounting-and.html

[45] Modern Spreadsheet, https://digitalfinancialreporting.blogspot.com/2023/05/modern-spreadsheet.html

[46] Universal Technology of Accountability, https://digitalfinancialreporting.blogspot.com/2023/02/universal-technology-for-accountability.html

[47] Charles Hoffman, CPA, *Logical Systems*, http://www.xbrlsite.com/mastering/Part02_Chapter05.A_LogicalSystems.pdf

simplistic; the example is not simplistic at all. This basic example shows the moving pieces of the puzzle clearly and the small size of the example helps the reader get their head around what is being described. As I will show using progressively larger and larger examples[48]; every base financial reporting scheme, every report model, and every report created by any reporting economic entity follows these exact same ideas and has similar moving pieces. Real logical systems will be larger and more complex and therefore more sophisticated but the basic moving pieces are exactly the same.

1.8. Very Simple Example of Logical System

We will provide an example of a logical system to strengthen your understanding of logical systems. A very simple example of a logical system is the accounting equation[49]. Here is a description of the accounting equation logical system in both human-readable terms and machine-readable terms using XBRL[50]:

Terms: Three simple terms are defined: Assets, Liabilities, Equity. One complex term is defined, balance sheet.

Structure: One structure is defined, the balance sheet, and identified using the term balance sheet.

Associations: The three terms Assets, Liabilities, and Equity are associated in that they are all PART-OF the structure balance sheet.

Rules: A mathematical assertion is made that "Assets = Liabilities + Equity".

Facts: Instances of three facts are established to exercise the model: Assets of $5,000; Liabilities of $1,000; Equity of $4,000.

Model: All of the terms, associations, rules, structures, and facts describe the model. We created only one model, or permissible interpretation, of the logical theory. (As accountants know, if you reverse the equation using the rules of math to "Equity = Assets - Liabilities" and change the term "Equity" to "Net Assets"; then you get another permissible interpretation or model.)

Because this is a very simple example with only a few statements it is easy to get your head around this system and see that it is consistent, complete, and precise. As expected, you see three facts described by three terms which are related to one structure and the one rule is consistent with expectation:

As the size of the logical system increases it becomes increasingly more challenging to verify the validity that the logical system is properly functioning using manual processes. We will discuss the types of things that can go wrong with a system in a later section. Understanding what can go wrong helps one understand the counter measures that need to be taken to make sure that nothing does go wrong.

Essentially, the models, terms, structures, rules, and facts form a directed acyclic graph or knowledge graph such as this simplified knowledge graph which describes the system we are discussing:
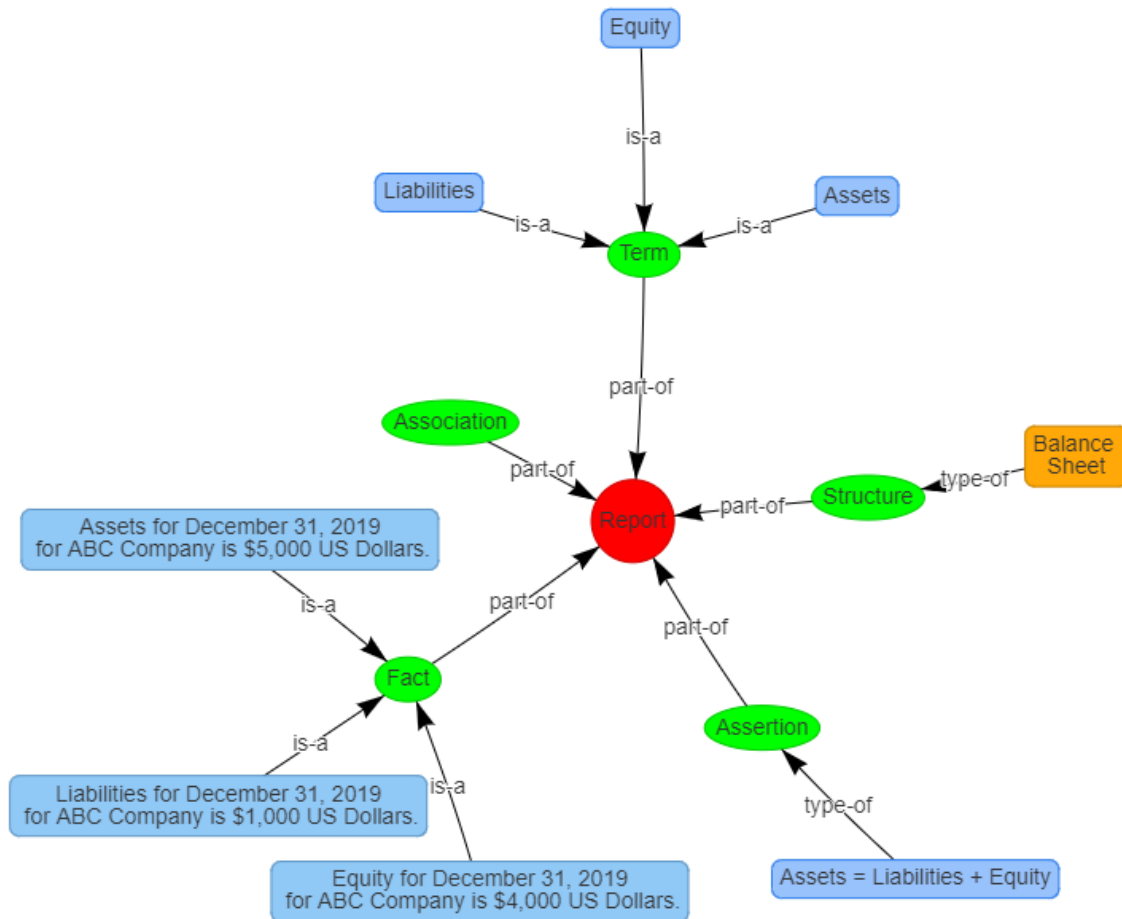
Hopefully you get the general idea from this simplified wireframe representation of our very basic accounting equation logical system as a knowledge graph.

While a typical financial report is significantly larger (i.e., the Microsoft 2017 10-K[51] is made up of 194 structures, 2035 facts, etc.) every financial report works the same as this very simple example but just has more pieces.

To engrain the basic ideas of a logical system and the financial report logical system in your mind, let us look at another very simple "hello world" example of a fragment of an XBRL-based digital financial report logically. This example will add additional

---

[51] Microsoft XBRL-based Report Analysis, http://xbrl.squarespace.com/journal/2020/4/13/microsoft-xbrl-based-report-analysis.html

details and will help you become better grounded in these basic, fundamental ideas that will help you understand XBRL-based digital financial reports better.

If you want more information about the financial report logical system at this point, I would encourage you to watch the YouTube.com video playlist *Understanding the Financial Report Logical System*[52].

2. Basics – Hello World!

Here we start with the very basics of XBRL-based digital financial reporting. In this section, please focus on the mechanics of what you are doing so that you understand the mechanical aspects of what you are doing.

In the next section we will build on your conceptual understanding.

We will work through the Hello World tutorial here in this section. If you have a software application for creating XBRL-based report models and reports and feel comfortable then move to the next section now. If you don't have software and don't feel comfortable; then please work through the Accounting Equation, SFAC 6, SFAC 8, Common Elements of Financial Report examples using the free Luca software application[53], the free open-source Microsoft Access database software application[54], or any software application of your choice. Work on these specific examples until you feel that you have the mechanical aspects of using your specific software application down pat.

These four videos will help you work through the mechanics of using the Luca or Microsoft Access database application:

- Accounting equation using Luca: https://youtu.be/KfT-uL-DIvE

- Accounting equation using Access: https://youtu.be/POPg49tp9PE

- SFAC 6 using Luca: https://youtu.be/O5TtuDPPuvQ

- SFAC 6 using Access: https://youtu.be/PZ_yYru4lgs

Again, the focus is on getting down the mechanics of creating a very basic XBRL-based digital financial report; understanding the terms model, structure, term, association, rules, and fact; and understanding the basics of how all these pieces fit together.

If you want to work through these examples using a cloud-based software application, try Auditchain Suite's Luca which is still under construction but works good enough to help you build report models and reports[55]. This video playlist[56] of what I believe is the world's first exert system for creating financial reports[57] will help you wrap your head around these ideas and become a master craftsmen.

---

[52] YouTube.com, Charles Hoffman, CPA, *Understanding the Financial Report Logical System*, https://www.youtube.com/playlist?list=PLqMZRUzQ64B7EWamzDP-WaYbS_W0RL9nt

[53] *Luca*, http://xbrl.squarespace.com/journal/2020/9/15/luca.html

[54] *Free Open Source Tool for Creating Quality XBRL-based Digital Financial Reports*, http://xbrl.squarespace.com/journal/2020/12/8/free-open-source-tool-for-creating-quality-xbrl-based-digita.html

[55] Auditchain Suite, https://dev.auditchain.finance/

[56] YouTube, World's First Expert system for Creating Financial Report (video playlist), https://www.youtube.com/playlist?list=PL80qjzvfqwtNuTekdlRy0rhaHEDIXkOh3

[57] World's First Expert System for Creating Financial Report (blog post), https://digitalfinancialreporting.blogspot.com/2023/01/worlds-first-standards-based-expert.html

To synthesize this understanding for you, we will walk through a very basic example next.

2.1. Cloud-based Luca

Another alternative to understanding the basics is to use a new cloud-based version of Luca[58]. Another cloud-based version of Luca is provided within the Auditchain Suite cloud-based software application[59].

2.2. Hello World! (logical perspective)

You have used a conceptual model if you are an accountant and probably don't even realize it. Electronic spreadsheets are broken down into the rows, columns, cells, sheets, and workbooks that make up the pieces of an electronic version of a spreadsheet. Those ideas came from the paper-based spreadsheet that likewise had rows, columns, cells, and sheets.
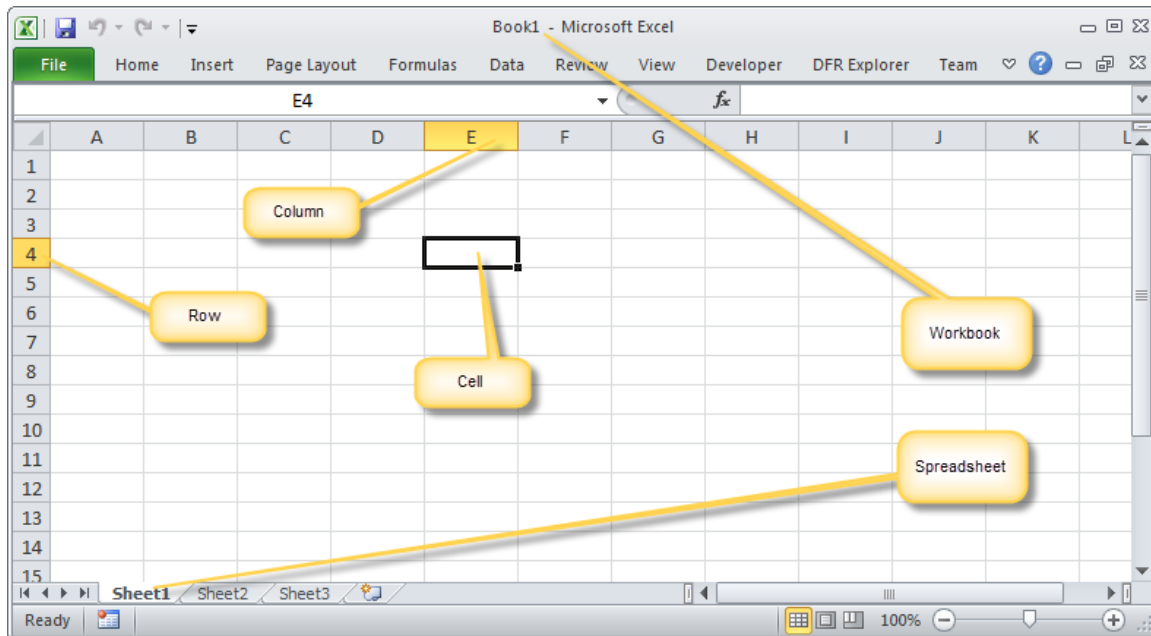
Did you realize that Excel is actually similar to the XBRL XML files? Create a workbook, rename the extension ".zip", and then go into the ZIP file. You will see a bunch of XML files. Do you ever work with Excel at the level of those XML files? Probably not. It is also highly-likely that you will never work with XBRL at the level of the XML files that holds the contents of the XBRL-based information either.

So, we will start with exactly the same screen shot of the same fragment of a financial report that we might use to look at the ugly technical syntax which software generates from the underlying XBRL file. Here is that human readable representation:

---

58 Cloud based Luca, http://xbrl.squarespace.com/journal/2021/8/31/cloud-based-luca.html
59 Auditchain Suite, https://suite.auditchain.finance/

Property, Plant and Equipment, Net [Roll Up]	Period [Axis]	
	2020-12-31	2019-12-31
Property, Plant and Equipment, Net [Roll Up]		
Land	5,347,000	1,147,000
Buildings, Net	244,508,000	366,375,000
Furniture and Fixtures, Net	34,457,000	34,457,000
Computer Equipment, Net	4,169,000	5,313,000
Other Property, Plant and Equipment, Net	6,702,000	6,149,000
Property, Plant and Equipment, Net	295,183,000	413,441,000

We enter the terms in a software application called Luca which you can download for free and use to create this Hello World example[60]. The Luca application uses a conceptual model, the *Logical Theory Describing Financial Report*[61], to hide the XBRL technical syntax from the software user, exposing only accounting logic which accountants understand. So, there is no need to understand the XBRL technical syntax.

Terms (simple terms)

From a logical perspective, we first define the simple terms that are used by the report model and the report itself. If you use Luca, after you have entered each term you see the following: (i.e. if you create this it should look exactly like the following)

Category	Standard Label	Prefix	Name	Data Type	Balance Type	Period Type
Abstract	Property, Plant and Equipment, Net [Roll Up]	helloWorld	PropertyPlantAndEquipmentNetRollUp			
Concept	Land	helloWorld	Land	Monetary	Debit	Instant
Concept	Buildings, Net	helloWorld	BuildingsNet	Monetary	Debit	Instant
Concept	Furniture and Fixtures, Net	helloWorld	FurnitureAndFixturesNet	Monetary	Debit	Instant
Concept	Computer Equipment, Net	helloWorld	ComputerEquipmentNet	Monetary	Debit	Instant
Concept	Other Property, Plant and Equipment, Net	helloWorld	OtherPropertyPlantAndEquipmentNet	Monetary	Debit	Instant
Concept	Property, Plant and Equipment, Net	helloWorld	PropertyPlantAndEquipmentNet	Monetary	Debit	Instant

Structures (complex terms)

From a logical perspective, we need to define structures using complex terms into which terms and association types will be used to explain the associations or relationships between terms. There is only one structure in the Hello World example: (again, what you create should look identical to what you see below)

---

[60] Luca, http://xbrl.squarespace.com/journal/2020/9/15/luca.html

[61] Logical Theory Describing Financial Report, http://xbrl.squarespace.com/logical-theory-financial-rep/

Structures

Identifier	Title	Sequence
HelloWorld	30000 - Property, Plant, and Equipment by Component	1

Associations

From a logical perspective, we must explicitly define the associations that enable a machine to understand how to organize facts and terms into the rendering that you see above. There are two sets of logical relations that we must express.

Presentation model structure

Associations

Structure Type	Network Identifier	Association From Name	Association Role	Association To Name	Calculation Polarity ▲	Preferred Label Role	Sequence
Presentation	HelloWorld	PropertyPlantAndEquipmentNetRollUp	Parent-Child	Land			1
Presentation	HelloWorld	PropertyPlantAndEquipmentNetRollUp	Parent-Child	BuildingsNet			2
Presentation	HelloWorld	PropertyPlantAndEquipmentNetRollUp	Parent-Child	FurnitureAndFixturesNet			3
Presentation	HelloWorld	PropertyPlantAndEquipmentNetRollUp	Parent-Child	ComputerEquipmentNet			4
Presentation	HelloWorld	PropertyPlantAndEquipmentNetRollUp	Parent-Child	OtherPropertyPlantAndEquipmentNet			5
Presentation	HelloWorld	PropertyPlantAndEquipmentNetRollUp	Parent-Child	PropertyPlantAndEquipmentNet			6

Mathematical roll up relations (one alternative approach for representing roll ups)

Associations

Structure Type	Network Identifier	Association From Name	Association Role	Association To Name	Calculation Polarity ▲	Preferred Label Role	Sequence
Calculation	HelloWorld	PropertyPlantAndEquipmentNet	Total-Item	Land	Add		1
Calculation	HelloWorld	PropertyPlantAndEquipmentNet	Total-Item	BuildingsNet	Add		2
Calculation	HelloWorld	PropertyPlantAndEquipmentNet	Total-Item	FurnitureAndFixturesNet	Add		3
Calculation	HelloWorld	PropertyPlantAndEquipmentNet	Total-Item	ComputerEquipmentNet	Add		4
Calculation	HelloWorld	PropertyPlantAndEquipmentNet	Total-Item	OtherPropertyPlantAndEquipmentNet	Add		5

Rules (optional, we already defined rule using XBRL calculations above)

From a logical perspective, we must explicitly define rules that enable a machine to understand required and/or permissible constraints, restrictions, derivation, assertions, and other such rules.

In this case, the only mathematical computation is the roll up of property, plant and equipment net which is expressed using XBRL calculation relations. We will duplicate this rule using a consistency rule:

Rules

Rule Type	Rule Code	Rule	Network	Concept	Sequence	Commentary
⊟ Consistency	BS1	$PropertyPlantAndEquipmentNet = $Land + $ BuildingsNet + $FurnitureAndFixturesNet + $ComputerEquipmentNet + $ OtherPropertyPlantAndEquipment	HelloWorld ▾	helloWorld:PropertyPlantAndEquipmentNet	1	
Variable Name						
helloWorld:PropertyPlantAndEquipmentNet						
helloWorld:Land						
helloWorld:BuildingsNet						
helloWorld:FurnitureAndFixturesNet						
▸ helloWorld:ComputerEquipmentNet						
helloWorld:OtherPropertyPlantAndEquipmentNet						

Facts

From a logical perspective, we must explicitly represent the facts that will be provided within the report. Here are those facts:

Reporting Entity Aspect	Calendar Period Aspect	ConceptAspect	Fact Value	Units	Rounding	Sequence
SAMP	http://www.reportingscheme.com/ID	2020-12-31	helloWorld:Land	5347000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2019-12-31	helloWorld:Land	1147000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2020-12-31	helloWorld:BuildingsNet	244508000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2019-12-31	helloWorld:BuildingsNet	366375000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2020-12-31	helloWorld:FurnitureAndFixturesNet	34457000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2019-12-31	helloWorld:FurnitureAndFixturesNet	34457000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2020-12-31	helloWorld:ComputerEquipmentNet	4169000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2019-12-31	helloWorld:ComputerEquipmentNet	5313000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2020-12-31	helloWorld:OtherPropertyPlantAndEquipmentNet	6702000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2019-12-31	helloWorld:OtherPropertyPlantAndEquipmentNet	6149000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2020-12-31	helloWorld:PropertyPlantAndEquipmentNet	295183000	iso4217:USD	INF	
SAMP	http://www.reportingscheme.com/ID	2019-12-31	helloWorld:PropertyPlantAndEquipmentNet	413441000	iso4217:USD	INF	

The end result is exactly the same as the technical perspective representation. The logical representation is used to generate the technical representation.

If you are interested, this link below has a Microsoft Access database application that generated all of the Hello World example files. If you are motivated, this is a great way to understand how to represent the conceptual model of a financial report within a relational database:

http://xbrlsite.azurewebsites.net/2020/master/hello-world-db/

The database generated application is a bit more sophisticated because it contains XBRL references and Inline XBRL examples. Software engineers trying to create financial reporting or business reporting tools that output XBRL might find this example useful.

After all the above is entered, saved as XBRL, verified to be consistent with the XBRL technical specification, and then loaded into a software application that understands XBRL you should see something like the following[62]:

Property, Plant and Equipment, Net [Roll Up]	Period [Axis]	
	2020-12-31	2019-12-31
Property, Plant and Equipment, Net [Roll Up]		
Land	5,347,000	1,147,000
Buildings, Net	244,508,000	366,375,000
Furnitures and Fixtures, Net	34,457,000	34,457,000
Computer Equipment, Net	4,169,000	5,313,000
Other Property, Plant and Equipment, Net	6,702,000	6,149,000
Property, Plant and Equipment, Net	295,183,000	413,441,000

Alternatively, if you load this into Pesseract, it would look like the following[63]:

---

[62] Hello world representation loaded into XBRL Cloud Evidence Package,
http://xbrlsite.azurewebsites.net/2020/master/hello-world-db/evidence-package/
[63] Pesseract, http://pesseract.azurewebsites.net

Reporting Entity [Axis]	SAMP http://www.reportingscheme.com/ID		
Unit [Axis]	USD		
	Period [Axis]		
Implied [Line Items]	2020-12-31	2019-12-31	
Property, Plant and Equipment, Net [Roll Up]			
Land	5,347,000	1,147,000	
Buildings, Net	244,508,000	366,375,000	
Furnitures and Fixtures, Net	34,457,000	34,457,000	
Computer Equipment, Net	4,169,000	5,313,000	
Other Property, Plant and Equipment, Net	6,702,000	6,149,000	
Property, Plant and Equipment, Net	295,183,000	413,441,000	

It may not seem obvious but what you are seeing is actually pretty remarkable. Two completely different software applications created by two completely different software vendors and the information looks virtually the same. Why is that? The reason is that XBRL is a global standard. Agreement as to the technical syntax and as to the logic of the representation means that any software application that supports the standard can use the reported information.

Not every detail is exactly the same, but important details are the same.

So, lets have a bit of a more detailed look what commercial software provides for the simple Hello World example that we created in XBRL.

2.3. Hello World! (commercial software perspective)

Now we are going to look at the exact same thing as that ugly technical stuff (that is coming in the next section) but we will look using a commercially available software tool provided by XBRL Cloud called the Evidence Package. We are using this because it is simply a set of HTML pages that shows you what is contained in those XBRL files in terms a professional accountant can understand. XBRL Cloud has let me make this and other examples available so that you can easily have a look at the examples that I am providing.

Now, in the explanation of the technical details of taxonomy schemas and linkbases I left a few details out to make the bigger picture easier to grasp. As I said, we will drill down into many of those details in this document. But to master all of the technical details will require more work that just reading this very basic primer.

But, let's go through what you will see in the next section with the technical details but this time we applying a logical conceptual model and see how much easier all this stuff is to read. Remember, what you see in the human readable representations was created from only the XBRL files that we created in the prior section. So here we go, preparing you to have a look at the technical details in the next section.

Terms

In the next section we will talk about the taxonomy schema and the label linkbase. Well, here that information is in a nicer computer-generated rendering[64]:

---

[64] Evidence Package, Terms, http://xbrlsite.azurewebsites.net/2020/master/hello-world/evidence-package/contents/index.html#ReportElementsAdded-Concepts.html

#	Label	Data Type	Period Type	Balance Type	Prefix	Standard label, Documentation, References, Concept name	Count
1	Buildings, Net	Monetary	As Of (instant)	Debit	helloWorld	*Filer label:* Buildings, Net *Documentation:* *References:* NONE *Name:* helloWorld:BuildingsNet	1
2	Computer Equipment, Net	Monetary	As Of (instant)	Debit	helloWorld	*Filer label:* Computer Equipment, Net *Documentation:* *References:* NONE *Name:* helloWorld:ComputerEquipmentNet	1
3	Furniture and Fixtures, Net	Monetary	As Of (instant)	Debit	helloWorld	*Filer label:* Furniture and Fixtures, Net *Documentation:* *References:* NONE *Name:* helloWorld:FurnitureAndFixturesNet	1
4	Land	Monetary	As Of (instant)	Debit	helloWorld	*Filer label:* Land *Documentation:* *References:* NONE *Name:* helloWorld:Land	1
5	Other Property, Plant and Equipment, Net	Monetary	As Of (instant)	Debit	helloWorld	*Filer label:* Other Property, Plant and Equipment, Net *Documentation:* *References:* NONE *Name:* helloWorld:OtherPropertyPlantAndEquipmentNet	1
6	Property, Plant and Equipment, Net	Monetary	As Of (instant)	Debit	helloWorld	*Filer label:* Property, Plant and Equipment, Net *Documentation:* *References:* NONE *Name:* helloWorld:PropertyPlantAndEquipmentNet	1

Rules

And, remember the mathematical relations represented by the XBRL calculation relations. Well, here are those in an easier to read form[65]:

Label	Rendered	Reported	Calculated	Balance	Decimals	Message
Property, Plant and Equipment, Net [Roll Up]						
Land	5,347,000 +	5,347,000	5,347,000	DR	INF	
Buildings, Net	244,508,000 +	244,508,000	244,508,000	DR	INF	
Furniture and Fixtures, Net	34,457,000 +	34,457,000	34,457,000	DR	INF	
Computer Equipment, Net	4,169,000 +	4,169,000	4,169,000	DR	INF	
Other Property, Plant and Equipment, Net	6,702,000 +	6,702,000	6,702,000	DR	INF	
Property, Plant and Equipment, Net	295,183,000	295,183,000	295,183,000	DR	INF	OK

Label	Rendered	Reported	Calculated	Balance	Decimals	Message
Property, Plant and Equipment, Net [Roll Up]						
Land	1,147,000 +	1,147,000	1,147,000	DR	INF	
Buildings, Net	366,375,000 +	366,375,000	366,375,000	DR	INF	
Furniture and Fixtures, Net	34,457,000 +	34,457,000	34,457,000	DR	INF	
Computer Equipment, Net	5,313,000 +	5,313,000	5,313,000	DR	INF	
Other Property, Plant and Equipment, Net	6,149,000 +	6,149,000	6,149,000	DR	INF	
Property, Plant and Equipment, Net	413,441,000	413,441,000	413,441,000	DR	INF	OK

Associations

Finally, here are the XBRL presentation relations represented in an easier to read form[66]:

Label	Report Element Class	Period Type	Balance	Name
Property, Plant and Equipment, Net [Roll Up]	[Abstract]			pattern:PropertyPlantAndEquipmentNetRollUp
Land	[Concept] Monetary	As Of	Debit	pattern:Land
Buildings, Net	[Concept] Monetary	As Of	Debit	pattern:BuildingsNet
Furniture and Fixtures, Net	[Concept] Monetary	As Of	Debit	pattern:FurnitureAndFixturesNet
Computer Equipment, Net	[Concept] Monetary	As Of	Debit	pattern:ComputerEquipmentNet
Other Property, Plant and Equipment, Net	[Concept] Monetary	As Of	Debit	pattern:OtherPropertyPlantAndEquipmentNet
Property, Plant and Equipment, Net, Total	[Concept] Monetary	As Of	Debit	pattern:PropertyPlantAndEquipmentNet

Report

---

[65] Evidence Package, Rules, http://xbrlsite.azurewebsites.net/2020/master/hello-world/evidence-package/contents/index.html#BusinessRulesSummary.html

[66] Evidence Package, Associations, http://xbrlsite.azurewebsites.net/2020/master/hello-world/evidence-package/contents/index.html#ModelSummary.html

A little more computer magic[67]; you take all those machine-readable details, run them through what is called an XBRL processor, and presto-chango; you get that nice looking, easy to read computer-generated rendering. And remember, the information is also readable by machine-based processes[68]:

Property, Plant and Equipment, Net [Roll Up]	Period [Axis]	
	2020-12-31	2019-12-31
Property, Plant and Equipment, Net [Roll Up]		
Land	5,347,000	1,147,000
Buildings, Net	244,508,000	366,375,000
Furniture and Fixtures, Net	34,457,000	34,457,000
Computer Equipment, Net	4,169,000	5,313,000
Other Property, Plant and Equipment, Net	6,702,000	6,149,000
Property, Plant and Equipment, Net	295,183,000	413,441,000

Again, remember that I left some of the details out to help provide an easy-to-understand introduction to XBRL. One commercial software vendor provides a static rendering and other reports that explains the information in an XBRL document that I can provide here for you to fiddle with[69]. If you want to dig into the XBRL technical syntax, here is a ZIP file that provides the actual XBRL that is behind the human readable reports[70].

2.4. Hello World! (OPTIONAL technical perspective)

Here is the technical view of this information that we were promising. If you don't like technical stuff, then skip this section. If you are curious about the technical details, read on.

Everyone these days knows what a hello world example is. We start our very basic primer with a *Hello World* example of XBRL. Consider this graphic below:

---

[67] YouTube.com, *How XBRL Works*, https://www.youtube.com/watch?v=nATJBPOiTxM (it really is not using magic)

[68] Evidence Package, Rendering, http://xbrlsite.azurewebsites.net/2020/master/hello-world/evidence-package/contents/index.html#Rendering-N0-Implied.html

[69] Hello World example human readable information, http://xbrlsite.azurewebsites.net/2018/HelloWorld/evidence-package/#Rendering-N0-RE10.html

[70] Hello World example machine readable XBRL, http://xbrlsite.azurewebsites.net/2018/HelloWorld/RollUp.zip

Property, Plant and Equipment, Net [Roll Up]	Period [Axis]	
	2020-12-31	2019-12-31
Property, Plant and Equipment, Net [Roll Up]		
Land	5,347,000	1,147,000
Buildings, Net	244,508,000	366,375,000
Furniture and Fixtures, Net	34,457,000	34,457,000
Computer Equipment, Net	4,169,000	5,313,000
Other Property, Plant and Equipment, Net	6,702,000	6,149,000
Property, Plant and Equipment, Net	295,183,000	413,441,000

What do you see? You see five items that add up to a total. Information is shown for two years. You can tell that this is a total because of the single and double underscores above and below the total value. The cell of the totals is green indicating that the items add up to the total.

The screen shot above was generated using XBRL. You can fiddle with that XBRL-based report using this rendering which was generated from the XBRL[71]. What we are going to do is explain to you how the information above was represented in XBRL. In order to enable a computer work with the information, we need to put the information into machine-readable form. We will use the XBRL technical syntax. XBRL is a type of XML[72]. Don't worry about that now, just know that XBRL is XML.

XBRL Taxonomy

The following is a taxonomy schema that is used as we begin to represent the information from that graphic in machine readable XBRL form.

```
<?xml version="1.0" encoding="utf-8"?>
<schema
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xbrli="http://www.xbrl.org/2003/instance"
  xmlns:helloWorld="http://www.xbrlsite.com/DigitalFinancialReporting/HelloWorld"
  targetNamespace="http://www.xbrlsite.com/DigitalFinancialReporting/HelloWorld">
  <element
    id="helloWorld_Land"
    name="Land"
    type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item"
    xbrli:periodType="instant" />
  <element
    id="helloWorld_BuildingsNet"
    name="BuildingsNet"
    type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item"
    xbrli:periodType="instant" />
  <element
    id="helloWorld_FurnitureAndFixturesNet"
    name="FurnitureAndFixturesNet"
    type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item"
    xbrli:periodType="instant" />
  <element
    id="helloWorld_ComputerEquipmentNet"
    name="ComputerEquipmentNet"
```

---

[71] Hello World example human readable rendering, http://xbrlsite.azurewebsites.net/2020/master/hello-world/evidence-package

[72] W3C, Extensible Markup Language (XML) 1.0 (Fifth Edition), https://www.w3.org/TR/xml/

```
        type="xbrli:monetaryItemType"
        substitutionGroup="xbrli:item"
        xbrli:periodType="instant" />
     <element
        id="helloWorld_OtherPropertyPlantAndEquipmentNet"
        name="OtherPropertyPlantAndEquipmentNet"
        type="xbrli:monetaryItemType"
        substitutionGroup="xbrli:item"
        xbrli:periodType="instant" />
     <element
        id="helloWorld_PropertyPlantAndEquipmentNet"
        name="PropertyPlantAndEquipmentNet"
        type="xbrli:monetaryItemType"
        substitutionGroup="xbrli:item"
        xbrli:periodType="instant" />

 </schema>
```

Note the "name" in the XBRL in what is called the taxonomy schema shown above. You can go to that actual XBRL taxonomy schema[73]. Note that there is an element which has a "name" for every line in the graphic on the prior page. You see elements with the name "Land", "BuildingsNet", "FurnitureAndFixturesNet", "ComputerEquipmentNet", "OtherPropertyPlandAndEquipmentNet", and "PropertyPlantAndEquipmentNet".

An XBRL taxonomy schema is an XML Schema[74].

Labels

Those names are sort of ugly to work with, so XBRL allows you to add labels and associate a label with a name. So, let's add labels to make the human readable rendering of the information easier to read. Those labels are created using what is called a linkbase[75]. Below you see a linkbase of labels, called an **XBRL label linkbase**:

```
 <?xml version="1.0" encoding="utf-8"?>
 <linkbase
   xmlns="http://www.xbrl.org/2003/linkbase"
   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xlink="http://www.w3.org/1999/xlink">
   <labelLink xlink:type="extended" xlink:role="http://www.xbrl.org/2003/role/link">
     <loc xlink:type="locator" xlink:label="helloWorld_BuildingsNet" xlink:href="HelloWorld.xsd#helloWorld_BuildingsNet" />
     <labelArc xlink:type="arc"  xlink:arcrole=http://www.xbrl.org/2003/arcrole/concept-label xlink:from="helloWorld_BuildingsNet"
 xlink:to="helloWorld_BuildingsNet_lbl" />
     <label xlink:type="resource" xlink:label="helloWorld_BuildingsNet_lbl" xlink:role="http://www.xbrl.org/2003/role/label">Buildings, Net</label>
     <loc xlink:type="locator"  xlink:label="helloWorld_ComputerEquipmentNet" xlink:href="HelloWorld.xsd#helloWorld_ComputerEquipmentNet" />
     <labelArc xlink:type="arc"   xlink:arcrole="http://www.xbrl.org/2003/arcrole/concept-label" xlink:from="helloWorld_ComputerEquipmentNet"
      xlink:to="helloWorld_ComputerEquipmentNet_lbl" />
     <label xlink:type="resource" xlink:label="helloWorld_ComputerEquipmentNet_lbl" xlink:role="http://www.xbrl.org/2003/role/label">Computer
 Equipment, Net</label>
     <loc xlink:type="locator"  xlink:label="helloWorld_FurnitureAndFixturesNet" xlink:href="HelloWorld.xsd#helloWorld_FurnitureAndFixturesNet"
 />
     <labelArc xlink:type="arc" xlink:arcrole="http://www.xbrl.org/2003/arcrole/concept-label" xlink:from="helloWorld_FurnitureAndFixturesNet"
 xlink:to="helloWorld_FurnitureAndFixturesNet_lbl" />
     <label xlink:type="resource" xlink:label="helloWorld_FurnitureAndFixturesNet_lbl" xlink:role="http://www.xbrl.org/2003/role/label">Furniture
 and Fixtures, Net</label>
     <loc xlink:type="locator"  xlink:label="helloWorld_Land"  xlink:href="HelloWorld.xsd#helloWorld_Land" />
     <labelArc xlink:type="arc"  xlink:arcrole="http://www.xbrl.org/2003/arcrole/concept-label"  xlink:from="helloWorld_Land"
 xlink:to="helloWorld_Land_lbl" />
     <label xlink:type="resource"  xlink:label="helloWorld_Land_lbl"  xlink:role="http://www.xbrl.org/2003/role/label">Land</label>
```

---

[73] Hello World XBRL Taxonomy Schema, http://xbrlsite.azurewebsites.net/2020/master/hello-world/HelloWorld.xsd

[74] W3C, W3C XML Schema Definition Language (XSD) 1.1 Part 1: Structures, https://www.w3.org/TR/xmlschema11-1/

[75] W3C, XML Linking Language (XLink) Version 1.1, https://www.w3.org/TR/xlink11/

```
    <loc xlink:type="locator"  xlink:label="helloWorld_OtherPropertyPlantAndEquipmentNet"
xlink:href="HelloWorld.xsd#helloWorld_OtherPropertyPlantAndEquipmentNet" />
    <labelArc xlink:type="arc"  xlink:arcrole="http://www.xbrl.org/2003/arcrole/concept-label"
xlink:from="helloWorld_OtherPropertyPlantAndEquipmentNet"
      xlink:to="helloWorld_OtherPropertyPlantAndEquipmentNet_lbl" />
    <label xlink:type="resource"  xlink:label="helloWorld_OtherPropertyPlantAndEquipmentNet_lbl"
xlink:role="http://www.xbrl.org/2003/role/label">Other Property, Plant and Equipment, Net</label>
    <loc xlink:type="locator"  xlink:label="helloWorld_PropertyPlantAndEquipmentNet"
xlink:href="HelloWorld.xsd#helloWorld_PropertyPlantAndEquipmentNet" />
    <labelArc xlink:type="arc"  xlink:arcrole="http://www.xbrl.org/2003/arcrole/concept-label"
xlink:from="helloWorld_PropertyPlantAndEquipmentNet"
      xlink:to="helloWorld_PropertyPlantAndEquipmentNet_lbl" />
    <label xlink:type="resource"  xlink:label="helloWorld_PropertyPlantAndEquipmentNet_lbl"
xlink:role="http://www.xbrl.org/2003/role/label">Property, Plant and Equipment, Net</label>
</labelLink>

</linkbase>
```

If you look closely at the information in the linkbase above, you can see the names of the concepts and you can see labels. The human readable labels are associated with the names of the taxonomy schema. Each name in the taxonomy schema has a human readable label associated with it in the XLink linkbase that is shown above. Here is the Hello World XBRL label linkbase[76].

Pretty ugly stuff, right? Well, it gets even uglier…because there are more linkbases.

XBRL calculations

Again, looking at the human readable rendering of the information you see that there are mathematical computations in the information that is being represented. In this case, there is a common type of mathematical computation called a roll up. XBRL uses XLink, just like for the labels, to articulate the mathematical computation, the roll up in this case, that the information we are trying to represent has. You can see the **XBRL calculation relations** below:

```
<?xml version="1.0" encoding="utf-8"?>
<linkbase
  xmlns="http://www.xbrl.org/2003/linkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink">
  <roleRef xlink:type="simple"
   xlink:href="HelloWorld.xsd#PropertyPlantAndEquipmentByComponent"
   roleURI="http://www.xbrlsite.com/DigitalFinancialReporting/HelloWorld/PropertyPlantAndEquipmentByComponent" />
  <calculationLink xlink:type="extended"
   xlink:role="http://www.xbrlsite.com/DigitalFinancialReporting/HelloWorld/PropertyPlantAndEquipmentByComponent">
   <loc xlink:type="locator"  xlink:label="helloWorld_PropertyPlantAndEquipmentNet"
    xlink:href="HelloWorld.xsd#helloWorld_PropertyPlantAndEquipmentNet" />
   <loc xlink:type="locator"  xlink:label="helloWorld_Land"  xlink:href="HelloWorld.xsd#helloWorld_Land" />
   <calculationArc xlink:type="arc"  xlink:arcrole="http://www.xbrl.org/2003/arcrole/summation-item"
    xlink:from="helloWorld_PropertyPlantAndEquipmentNet"  xlink:to="helloWorld_Land"  order="1"  weight="1"  use="optional" />
   <loc xlink:type="locator" xlink:label="helloWorld_BuildingsNet" xlink:href="HelloWorld.xsd#helloWorld_BuildingsNet" />
   <calculationArc            xlink:type="arc"                            xlink:arcrole="http://www.xbrl.org/2003/arcrole/summation-item"
xlink:from="helloWorld_PropertyPlantAndEquipmentNet"
    xlink:to="helloWorld_BuildingsNet"  order="2"  weight="1"  use="optional" />
   <loc xlink:type="locator"  xlink:label="helloWorld_FurnitureAndFixturesNet"  xlink:href="HelloWorld.xsd#helloWorld_FurnitureAndFixturesNet" />
   <calculationArc            xlink:type="arc"                            xlink:arcrole="http://www.xbrl.org/2003/arcrole/summation-item"
xlink:from="helloWorld_PropertyPlantAndEquipmentNet"
    xlink:to="helloWorld_FurnitureAndFixturesNet"  order="3"  weight="1"  use="optional" />
   <loc xlink:type="locator"  xlink:label="helloWorld_ComputerEquipmentNet"  xlink:href="HelloWorld.xsd#helloWorld_ComputerEquipmentNet" />
   <calculationArc            xlink:type="arc"                            xlink:arcrole="http://www.xbrl.org/2003/arcrole/summation-item"
xlink:from="helloWorld_PropertyPlantAndEquipmentNet"
    xlink:to="helloWorld_ComputerEquipmentNet"  order="4"  weight="1"  use="optional" />
```

---

[76] Hello World XBRL label linkbase, http://xbrlsite.azurewebsites.net/2020/master/hello-world/HelloWorld-label.xml

```
    <loc                      xlink:type="locator"                              xlink:label="helloWorld_OtherPropertyPlantAndEquipmentNet"
xlink:href="HelloWorld.xsd#helloWorld_OtherPropertyPlantAndEquipmentNet" />
    <calculationArc            xlink:type="arc"                                  xlink:arcrole="http://www.xbrl.org/2003/arcrole/summation-item"
xlink:from="helloWorld_PropertyPlantAndEquipmentNet"
    xlink:to="helloWorld_OtherPropertyPlantAndEquipmentNet" order="5"  weight="1"  use="optional" />
  </calculationLink>

</linkbase>
```

Here is the XBRL calculations linkbase that you see above[77].

XBRL presentation relations

Unfortunately, there is one other ugly set of XLink linkbase relations we need to show you. Notice that the lines in that graphic of the property, plant, and equipment roll up are in a specific order. **XBRL presentation relations** are used to describe the ordering of the information you want to represent.

```
<?xml version="1.0" encoding="utf-8"?>
<linkbase
  xmlns="http://www.xbrl.org/2003/linkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink">
  <roleRef xlink:type="simple" xlink:href="HelloWorld.xsd#PropertyPlantAndEquipmentByComponent"
roleURI="http://www.xbrlsite.com/DigitalFinancialReporting/HelloWorld/PropertyPlantAndEquipmentByComponent" />
  <presentationLink xlink:type="extended"
xlink:role="http://www.xbrlsite.com/DigitalFinancialReporting/HelloWorld/PropertyPlantAndEquipmentByComponent">
    <loc xlink:type="locator" xlink:label="helloWorld_Land" xlink:href="HelloWorld.xsd#helloWorld_Land" />
    <presentationArc xlink:type="arc" xlink:arcrole="http://www.xbrl.org/2003/arcrole/parent-child"
xlink:from="helloWorld_PropertyPlantAndEquipmentNetRollUp" xlink:to="helloWorld_Land"  order="1" use="optional" />
    <loc xlink:type="locator" xlink:label="helloWorld_BuildingsNet" xlink:href="HelloWorld.xsd#helloWorld_BuildingsNet" />
    <presentationArc xlink:type="arc" xlink:arcrole="http://www.xbrl.org/2003/arcrole/parent-child"
xlink:from="helloWorld_PropertyPlantAndEquipmentNetRollUp" xlink:to="helloWorld_BuildingsNet" order="2" use="optional" />
    <loc xlink:type="locator" xlink:label="helloWorld_FurnitureAndFixturesNet" xlink:href="HelloWorld.xsd#helloWorld_FurnitureAndFixturesNet" />
    <presentationArc xlink:type="arc" xlink:arcrole="http://www.xbrl.org/2003/arcrole/parent-child"
xlink:from="helloWorld_PropertyPlantAndEquipmentNetRollUp" xlink:to="helloWorld_FurnitureAndFixturesNet" order="3" use="optional" />
    <loc xlink:type="locator" xlink:label="helloWorld_ComputerEquipmentNet" xlink:href="HelloWorld.xsd#helloWorld_ComputerEquipmentNet" />
    <presentationArc xlink:type="arc" xlink:arcrole="http://www.xbrl.org/2003/arcrole/parent-child"
xlink:from="helloWorld_PropertyPlantAndEquipmentNetRollUp" xlink:to="helloWorld_ComputerEquipmentNet" order="4" use="optional" />
    <loc xlink:type="locator" xlink:label="helloWorld_OtherPropertyPlantAndEquipmentNet"
xlink:href="HelloWorld.xsd#helloWorld_OtherPropertyPlantAndEquipmentNet" />
    <presentationArc xlink:type="arc" xlink:arcrole="http://www.xbrl.org/2003/arcrole/parent-child"
xlink:from="helloWorld_PropertyPlantAndEquipmentNetRollUp" xlink:to="helloWorld_OtherPropertyPlantAndEquipmentNet" order="5"
use="optional" />
    <loc xlink:type="locator" xlink:label="helloWorld_PropertyPlantAndEquipmentNet"
xlink:href="HelloWorld.xsd#helloWorld_PropertyPlantAndEquipmentNet" />
    <presentationArc xlink:type="arc" xlink:arcrole="http://www.xbrl.org/2003/arcrole/parent-child"
xlink:from="helloWorld_PropertyPlantAndEquipmentNetRollUp" xlink:to="helloWorld_PropertyPlantAndEquipmentNet"
preferredLabel="http://www.xbrl.org/2003/role/totalLabel" order="6" use="optional" />
  </presentationLink>

</linkbase>
```

Here are the XBRL presentation linkbase that you see above[78].

There is another linkbase, the **XBRL definition relations** linkbase which is more advanced in nature, we will explain that and a few other linkbases later.

XBRL instance

---

[77] Hello World XBRL calculation linkbase, http://xbrlsite.azurewebsites.net/2020/master/hello-world/HelloWorld-calculation.xml
[78] Hello World presentation linkbase, http://xbrlsite.azurewebsites.net/2020/master/hello-world/HelloWorld-presentation.xml

Finally, we look at the XBRL instance. An XBRL instance contains the information for the report itself. Everything we were looking at previously is used to represent the model of the report. We will explain models if future sections.

```xml
<?xml version="1.0" encoding="utf-8"?>
<xbrl xmlns="http://www.xbrl.org/2003/instance"
    xmlns:xbrli="http://www.xbrl.org/2003/instance"
    xmlns:link="http://www.xbrl.org/2003/linkbase"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:helloWorld="http://www.xbrlsite.com/DigitalFinancialReporting/HelloWorld"
    xmlns:iso4217="http://www.xbrl.org/2003/iso4217"
    xsi:schemaLocation="http://xbrl.org/2006/xbrldi http://www.xbrl.org/2006/xbrldi-2006.xsd">

  <link:schemaRef xlink:type="simple" xlink:href="HelloWorld.xsd" />
  <context id="I-2020">
    <entity>
      <identifier scheme="http://www.SampleCompany.com">SAMP</identifier>
    </entity>
    <period>
      <instant>2020-12-31</instant>
    </period>
  </context>
  <context id="I-2019">
    <entity>
      <identifier scheme="http://www.SampleCompany.com">SAMP</identifier>
    </entity>
    <period>
      <instant>2019-12-31</instant>
    </period>
  </context>
  <unit id="U-Monetary">
    <measure>iso4217:USD</measure>
  </unit>

  <helloWorld:Land contextRef="I-2020" unitRef="U-Monetary" decimals="INF">5347000</helloWorld:Land>
  <helloWorld:Land contextRef="I-2019" unitRef="U-Monetary" decimals="INF">1147000</helloWorld:Land>
  <helloWorld:BuildingsNet contextRef="I-2020" unitRef="U-Monetary" decimals="INF">244508000</helloWorld:BuildingsNet>
  <helloWorld:BuildingsNet contextRef="I-2019" unitRef="U-Monetary" decimals="INF">366375000</helloWorld:BuildingsNet>
  <helloWorld:FurnitureAndFixturesNet contextRef="I-2020" unitRef="U-Monetary"
decimals="INF">34457000</helloWorld:FurnitureAndFixturesNet>
  <helloWorld:FurnitureAndFixturesNet contextRef="I-2019" unitRef="U-Monetary"
decimals="INF">34457000</helloWorld:FurnitureAndFixturesNet>
  <helloWorld:ComputerEquipmentNet contextRef="I-2020" unitRef="U-Monetary"
decimals="INF">4169000</helloWorld:ComputerEquipmentNet>
  <helloWorld:ComputerEquipmentNet contextRef="I-2019" unitRef="U-Monetary"
decimals="INF">5313000</helloWorld:ComputerEquipmentNet>
  <helloWorld:OtherPropertyPlantAndEquipmentNet contextRef="I-2020" unitRef="U-Monetary"
decimals="INF">6702000</helloWorld:OtherPropertyPlantAndEquipmentNet>
  <helloWorld:OtherPropertyPlantAndEquipmentNet contextRef="I-2019" unitRef="U-Monetary"
decimals="INF">6149000</helloWorld:OtherPropertyPlantAndEquipmentNet>
  <helloWorld:PropertyPlantAndEquipmentNet contextRef="I-2020" unitRef="U-Monetary"
decimals="INF">295183000</helloWorld:PropertyPlantAndEquipmentNet>
  <helloWorld:PropertyPlantAndEquipmentNet contextRef="I-2019" unitRef="U-Monetary"
decimals="INF">413441000</helloWorld:PropertyPlantAndEquipmentNet>

</xbrl>
```

Here is the XBRL instance that you see above[79].

Putting Everything Together

---

[79] Hello World XBRL instance, http://xbrlsite.azurewebsites.net/2020/master/hello-world/HelloWorld-SampleInstance.xml

The XBRL instance references the XBRL taxonomy. You can see the same names in the XBRL instance as was in the XBRL taxonomy schema. We will explain the technical details later.

What we want to point out is that when a computer loads the XBRL instance; the XBRL instance points to the XBRL taxonomy schema. The XBRL taxonomy schema points to each of the linkbases associated with the model and will load those. All of this happens automatically and is part of the XBRL technical specification and how software is supposed to process the XBRL.

So, here are all the XBRL files and other information for the example above which you can download and have a look at:

http://xbrlsite.azurewebsites.net/2020/master/hello-world/

So, those are the ugly technical details. There is great news! Professional accountants will never have to deal with those ugly technical details. Why? Because there is this other technical thing called a conceptual model.

We hope you enjoyed the technical information, but let us get back to an XBRL-based digital financial report using a more understandable logical perspective.

If you feel you want more technical information, I would suggest two resources. The first is a *Very Basic XBRL Technical Primer*[80] which is about 40 pages. The second is *The XBRL Book: Simple, precise, technical*[81] which is about 350 pages and can be obtained from Amazon.com should you feel you want to make that investment.

Once you have the mechanics of using software down, let us move to understanding the logical conceptualization of a financial report better which we will do now.

3. Logical Conceptualization of a Financial Report

In this section we will walk you through a narrative of the logical conceptualization of a financial report. In this section, please focus on the conceptual ideas even though you might not be able to implement these ideas using a software application yet.

The first question you might have is how did we come up with the logical conceptualization. Well, we did that by looking at a bunch of XBRL-based financial reports and we looked for patterns. We worked out the details, gave things names, and condensed what we saw into a model which we documented in the *Logical Theory Describing Financial Report*[82].

Now, you don't need to understand that theory in detail. Understanding all the details can be helpful; but most of the details are for software engineers that need to build software applications that provide the nice looking human readable information that we have shown.

---

[80] Charles Hoffman, CPA, Very Basic XBRL Technical Primer, http://www.xbrlsite.com/mastering/Part00_Chapter01.B_XBRLPrimer.pdf

[81] Ghislain Fourny, *The XBRL Book: Simple, precise, technical*, https://www.amazon.com/dp/B08RQZJ6VK

[82] Logical Theory Describing Financial Report, http://xbrl.squarespace.com/logical-theory-financial-rep/

3.1. Breaking Down the Pieces of an XBRL-based Digital Financial Report

First, we had to figure out the logical pieces of what is in XBRL-based financial reports. The information below is for a set of 6,023 XBRL-based financial reports submitted to the SEC by public companies[83]:

- Total reports: 6,023
- Total facts reported: 8,532,275
- *Average facts per report*: 1,416
- Total networks in all reports: 462,786
- *Average networks per report*: 77
- Total fact sets in all reports: 754,430
- *Average fact sets per report*: 125
- Average fact sets per network: 1.6
- *Average facts per network*: 18
- *Average facts per fact set*: 11

Of the 754,430 fact sets there are:

- Text blocks: 407,392 (54%) are text blocks (Level 1 Notes, Level 2 Policies, Level 3 Disclosures)
- Sets: 181,063 (24%) are sets (or hierarchies, no mathematical computations)
- Roll ups: 120,708 (16%) are roll ups
- Roll forwards: 37,721 (5%) are roll forwards
- Roll forward info: 7,546 (1%) are roll forward infos or something else

Rather than simply working with a report, software can interact with the primitive pieces and the functional or complex pieces of that report with good affect[84].

3.2. Establishing the Financial Report Logical Conceptualization

And so, every XBRL-based financial report submitted to the SEC using US GAAP or IFRS, the ESMA field tests[85], and all my prototypes[86] fit into my one common meta-meta-model. How do I know this? Because I loaded 100% of US GAAP and IFRS XBRL-based reports, ESMA field tests, and all my prototypes into my model[87] to be

---

[83] *Breaking Down the Pieces of an XBRL-based Digital Financial Report*, http://xbrl.squarespace.com/journal/2019/4/9/breaking-down-the-pieces-of-an-xbrl-based-digital-financial.html

[84] *Leveraging Functional Components for XBRL-based Digital Financial Reporting*, http://xbrlsite.azurewebsites.net/2019/Library/LeveragingFunctionalComponents.pdf

[85] ESMA Field Tests, http://xbrl.squarespace.com/journal/2018/7/10/esma-field-test-information-great-information-for-testing.html

[86] *Mastering XBRL-based Digital Financial Reporting*, http://xbrlsite.azurewebsites.net/2020/master/

[87] *Understanding Digital*, Page 46, http://xbrlsite.azurewebsites.net/2020/Library/UnderstandingDigital.pdf#page=46

certain that my model was correct (i.e. test the model). What you are seeing is effectively the end result of all that work.

Here are the results from one of those tests:

So, the actual average size of the pieces of a report are quite small. Information is skewed a bit by the relatively large number of Level 1, Level 2, and Level 3 text blocks. Of the **754,430** fact sets there are:

- **Text Blocks**: 407,392 (54%) are text blocks (Level 1 Notes, Level 2 Policies, Level 3 Disclosures)
- **Sets**: 181,063 (24%) are sets (or hierarchies, no mathematical computations)
- **Roll Ups**: 120,708 (16%) are roll ups
- **Roll Forwards**: 37,721 (5%) are roll forwards
- Other (including **Roll Forward Info**, **Adjustment**, **Variance**): 7,546 (1%) are Roll Forward Infos or something else

What is more, which we will get to elsewhere in these documents, each of the concept arrangement

And so, you should be able to see the connections between the information from the 754,430 fact sets (a.k.a. Blocks) from above and the "Pattern" column in the PROOF representation which we will get to later and is used to TEST the meta-meta-model to see if our model provides everything that is necessary below. (Note that "Hierarchy" and "Set" are synonyms).

Now we will walk you through the logical conceptualization of a financial report in a manner that is approachable to professional accountants. Buckle up!

3.3. Scalar

A scalar is a fact which has no characteristics; it stands on its own. For example, the value of pi is a scalar, the value never changes; it always has the same value for everyone. (Pi or π is the ratio of a circle's circumference to its diameter and always has the value of equal to 3.14)

Fact Value
3.14

3.4. Facts

A financial **report**[88] communicates facts. A **fact**[89] defines a single, observable, reportable piece of information contained within a financial report, or **fact value**[90], contextualized for unambiguous interpretation or analysis by one or more distinguishing **aspects** (a.k.a characteristics). For example, below are two facts with the values of "2,000" and "1,000". However, the two facts above are not contextualized.

Fact Value
2,000
1,000

---

[88] Report, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Report.html

[89] Fact, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Fact.html

[90] Fact Value, http://xbrlsite.azurewebsites.net/2019/Framework/Details/FactValue.html

3.5. Aspects (a.k.a. Dimensions)

An **aspect**[91] describes a fact. An aspect provides information necessary to describe a fact or distinguish one fact from another fact within a report. For example, below you see the concept aspect of the numbers "2,000" and "1,000" which relates to the concepts "Revenues" and "Net income" respectively:

Concept Aspect	Fact Value
Revenues	2,000
Net income	1,000

To fully describe a fact, you need more than just one aspect. In XBRL-based financial reports, a fact must always provide three **core aspects**[92]: reporting entity that reported the fact, calendar period of the reported fact, and the concept that describes the reported fact. Below you see two facts which are characterized by three core aspects which are used to differentiate the two facts from one another.

Reporting Entity Aspect	Calendar Period Aspect	Concept Aspect	Fact Value
ABC Company	Jan 1, 2019 to Dec 31, 2019	Revenues	2,000
ABC Company	Jan 1, 2019 to Dec 31, 2019	Net income	1,000

In XBRL-based financial reports, in addition to the core aspects that you always must use, creators of reports can also provide additional **noncore aspects**[93]. A noncore aspect is simply some additional aspect that is created to further distinguish facts beyond the capabilities of the three core aspects.

Below you see the noncore aspect "Legal Entity Aspect" has been added to the two facts we have been working with:

Reporting Entity Aspect	Legal Entity Aspect	Calendar Period Aspect	Concept Aspect	Fact Value
ABC Company	Consolidated entity	Jan 1, 2019 to Dec 31, 2019	Revenues	2,000
ABC Company	Consolidated entity	Jan 1, 2019 to Dec 31, 2019	Net income	1,000

Fact values can be **numeric**[94] or **nonnumeric**[95]. Numeric fact values require additional information to describe the units of the numeric fact and the rounding that is used to report the numeric fact. **Units**[96] and **rounding**[97] are properties of the fact value that provide information necessary to describe numeric fact values.

Below you see that the units of "US Dollars" and that the rounding of "Thousands of dollars":

---

[91] Aspect, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Aspect.html

[92] Core Aspect, http://xbrlsite.azurewebsites.net/2019/Framework/Details/CoreAspect.html

[93] Noncore Aspect, http://xbrlsite.azurewebsites.net/2019/Framework/Details/NoncoreAspect.html

[94] Numeric Fact Value, http://xbrlsite.azurewebsites.net/2019/Framework/Details/NumericFactValue.html

[95] Nonnumeric Fact Value, http://xbrlsite.azurewebsites.net/2019/Framework/Details/NonnumericFactValue.html

[96] Units, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Units.html

[97] Rounding, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Rounding.html

Reporting Entity Aspect	Legal Entity Aspect	Calendar Period Aspect	Concept Aspect	Fact Value	Units	Rounding
ABC Company	Consolidated entity	Jan 1, 2019 to Dec 31, 2019	Revenues	2,000	US Dollars	Thousands of dollars
ABC Company	Consolidated entity	Jan 1, 2019 to Dec 31, 2019	Net income	1,000	US Dollars	Thousands of dollars

To summarize where we are thus far and to be crystal clear; below you see one fact. That single fact is characterized by a set of four aspects. The numeric fact value is described as having units of "US Dollars" and that the fact value is rounded to the nearest "Thousands of dollars".

Reporting Entity Aspect	Legal Entity Aspect	Calendar Period Aspect	Concept Aspect	Fact Value	Units	Rounding
ABC Company	Consolidated entity	Jan 1, 2019 to Dec 31, 2019	Revenues	2,000	US Dollars	Thousands of dollars

3.6. Information Block (a.k.a. Fact Set)

An **information block**[98] or simply block is a set of facts which go together (tend to be cohesive and share a certain common nature) for some specific purpose within a financial report.

Below you see three facts that make up an information block that are used to describe the breakdown of revenues by geographic area.

Reporting Entity Aspect	Legal Entity Aspect	Geographic Area Aspect	Calendar Period Aspect	Concept Aspect	Fact Value	Units	Rounding
ABC Company	Consolidated entity	All Geographic Areas Combined	Jan 1, 2019 to Dec 31, 2019	Revenues	2,000	US Dollars	Thousands of dollars
ABC Company	Consolidated entity	North America	Jan 1, 2019 to Dec 31, 2019	Revenues	1,000	US Dollars	Thousands of dollars
ABC Company	Consolidated entity	South America	Jan 1, 2019 to Dec 31, 2019	Revenues	1,000	US Dollars	Thousands of dollars

3.7. Rules (a.k.a. assertions, restrictions, constraints)

Rules[99] guide, control, suggest, or influence behavior. Rules cause things to happen, prevent things from happening, or suggest that it might be a good idea if something did or did not happen. Rules help shape judgment, help make decisions, help evaluate, help shape behavior.

Don't make the mistake of thinking that rules are completely inflexible and that you cannot break rules. Sure, maybe there are some rules that can never be broken. Maybe there are some rules that you can break. It helps to think of breaking rules as penalties in a football game. The point is that the guidance, control, suggestions, and influence offered by rules are a choice of business professionals. The meaning of a rule is separate from the level of enforcement someone might apply to the rule.

---

[98] Fact Set, http://xbrlsite.azurewebsites.net/2019/Framework/Details/FactSet.html

[99] Rule, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Rule.html

So, considering our fact set below we know that the value "2,000" is for the concept "Revenues", for the period "Jan 1, 2019 to Dec 31, 2019", relates to the legal entity "Consolidated entity", of the reporting entity "ABC Company" and is the total of all "Geographic Areas". "North America" and "South America" are part of the *whole* "All Geographic Areas Combined". A rule that expresses that relationship might be expressed as:

"All Geographic Areas Combined = North America + South America".

Rules both describe and can be used to verify that reported facts are consistent with the provided description. There are many different types of rules including mathematical, structural, mechanical, logical, and accounting related rules.

Reporting Entity Aspect	Legal Entity Aspect	Geographic Area Aspect	Calendar Period Aspect	Concept Aspect	Fact Value	Units	Rounding
ABC Company	Consolidated entity	All Geographic Areas Combined	Jan 1, 2019 to Dec 31, 2019	Revenues	2,000	US Dollars	Thousands of dollars
ABC Company	Consolidated entity	North America	Jan 1, 2019 to Dec 31, 2019	Revenues	1,000	US Dollars	Thousands of dollars
ABC Company	Consolidated entity	South America	Jan 1, 2019 to Dec 31, 2019	Revenues	1,000	US Dollars	Thousands of dollars

3.8. Granularity

Grain[100] is the level of depth of information or granularity. The lowest level of granularity is the actual transaction, event, circumstance, or other phenomenon represented as the actual transaction within an accounting system. The highest level of granularity is the summarized information that is represented as a line item of say the income statement.

Considering the information block you see below the fact outlined in red is one level of granularity as contrast to the other two facts that are outlined in green which provides the same information as is provided by the fact outlined in red, but at a different level of granularity.

Reporting Entity Aspect	Legal Entity Aspect	Geographic Area Aspect	Calendar Period Aspect	Concept Aspect	Fact Value	Units	Rounding
ABC Company	Consolidated entity	All Geographic Areas Combined	Jan 1, 2019 to Dec 31, 2019	Revenues	2,000	US Dollars	Thousands of dollars
ABC Company	Consolidated entity	North America	Jan 1, 2019 to Dec 31, 2019	Revenues	1,000	US Dollars	Thousands of dollars
ABC Company	Consolidated entity	South America	Jan 1, 2019 to Dec 31, 2019	Revenues	1,000	US Dollars	Thousands of dollars

And so hopefully you get an idea of the logical model of a financial report. Now we want to shift gears a bit and be a bit more specific as to how financial reports are represented using XBRL.

---

[100] Grain, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Grain.html

3.9. *Information Model Definition (a.k.a. Model)*

An **information model definition**[101] is created to represent each fragment of a report using the XBRL format. The following pieces, or **report elements**[102], are used to construct the information model description: **Network**[103], **Hypercube** (a.k.a. Table)[104], **Dimension** (a.k.a. Axis)[105], **Member**[106], **Line Items**[107], **Abstract**[108], and **Concept**[109].

Below you see the information model description of the structure of a fragment of a report, in this case one fact set which is used to describe the components of inventory:

#	Label	Report Element Class	Period Type	Balance	Name
1	**Inventory, by Compoment [Table]**	[Table]			gaap:InventoryByCompomentTable
2	**Legal Entity [Axis]**	[Axis]			frm:LegalEntityAxis
3	Consolidated Entity [Member]	[Member]			frm:ConsolidatedEntityMember
4	**Inventory, by Component [Line Items]**	[Line Items]			gaap:InventoryByComponentLineItems
5	Inventory, by Component [Roll Up]	[Abstract]			gaap:InventoryByComponentRollUp
6	Finished Goods	[Concept] Monetary	As Of	Debit	gaap:FinishedGoods
7	Work in Progress	[Concept] Monetary	As Of	Debit	gaap:WorkInProgress
8	Raw Material	[Concept] Monetary	As Of	Debit	gaap:RawMaterial
9	Inventory	[Concept] Monetary	As Of	Debit	gaap:Inventory

Something is important to point out. We mentioned that in XBRL you have core aspects and noncore aspects. In the typical software application created today, the core aspects reporting entity and calendar period are not represented in the information model description that is typically created by software applications.

Below you see a truer information model description which includes the reporting entity and the calendar period. Also, per the US GAAP XBRL Taxonomy, the IFRS XBRL Taxonomy the term "[Axis]" is used as a synonym of "Aspect". Axis and aspect are synonyms and mean the same thing. Also "Period" and "Calendar Period" are the same thing.

---

[101] Information Model Definition, http://xbrlsite.azurewebsites.net/2019/Framework/Details/InformationModelDefinition.html

[102] Report Element, http://xbrlsite.azurewebsites.net/2019/Framework/Details/ReportElement.html

[103] Network, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Network.html

[104] Table, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Table.html

[105] Axis, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Axis.html

[106] Member, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Member.html

[107] Line Items, http://xbrlsite.azurewebsites.net/2019/Framework/Details/LineItems.html

[108] Abstract, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Abstract.html

[109] Concept, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Concept.html

#	Label	Report Element Class	Period Type	Balance	Name
1	Inventory, by Compoment [Table]	[Table]			gaap:InventoryByCompomentTable
2	Reporting Entity [Axis]	[Axis]			xbrl:ReportingEntityAxis
3	http://regulator.gov/id#1234567890	[Member]			http://regulator.gov/id#1234567890
4	Period [Axis]	[Axis]			xbrl:PeriodAxis
5	12/31/2010	[Member]			12/31/2010
6	12/31/2009	[Member]			12/31/2009
7	Legal Entity [Axis]	[Axis]			frm:LegalEntityAxis
8	Consolidated Entity [Member]	[Member]			frm:ConsolidatedEntityMember
9	Inventory, by Component [Line Items]	[Line Items]			gaap:InventoryByComponentLineItems
10	Inventory, by Component [Roll Up]	[Abstract]			gaap:InventoryByComponentRollUp
11	Finished Goods	[Concept] Monetary	As Of	Debit	gaap:FinishedGoods
12	Work in Progress	[Concept] Monetary	As Of	Debit	gaap:WorkInProgress
13	Raw Material	[Concept] Monetary	As Of	Debit	gaap:RawMaterial
14	Inventory	[Concept] Monetary	As Of	Debit	gaap:Inventory

Another part of the information model description is the mathematical rules that are used to describe and verify the roll up relations of the concepts that are a part of the information model description. Here is the roll up relations that are part of this information model description.

#	Label	Report Element Class	Weight	Balance	Name
1	Inventory	[Concept] Monetary			gaap:Inventory
2	Finished Goods	[Concept] Monetary	+1	Debit	gaap:FinishedGoods
3	Work in Progress	[Concept] Monetary	+1	Debit	gaap:WorkInProgress
4	Raw Material	[Concept] Monetary	+1	Debit	gaap:RawMaterial

Another part of the information model description is the facts within the information block themselves. Here is the **fact table**[110] for the information block for the facts that go with the information model description provided above.

#	Reporting Entity [Aspect]	Period [Aspect]	Concept [Aspect]	Legal Entity [Aspect]	Fact Value	Unit	Rounding
1	http://regulator.gov/id#1234567890	12/31/2010	Finished Goods	Consolidated Entity [Member]	600,000	USD	Thousands
2	http://regulator.gov/id#1234567890	12/31/2009	Finished Goods	Consolidated Entity [Member]	600,000	USD	Thousands
3	http://regulator.gov/id#1234567890	12/31/2010	Work in Progress	Consolidated Entity [Member]	300,000	USD	Thousands
4	http://regulator.gov/id#1234567890	12/31/2009	Work in Progress	Consolidated Entity [Member]	300,000	USD	Thousands
5	http://regulator.gov/id#1234567890	12/31/2010	Raw Material	Consolidated Entity [Member]	100,000	USD	Thousands
6	http://regulator.gov/id#1234567890	12/31/2009	Raw Material	Consolidated Entity [Member]	100,000	USD	Thousands
7	http://regulator.gov/id#1234567890	12/31/2010	Inventory	Consolidated Entity [Member]	1,000,000	USD	Thousands
8	http://regulator.gov/id#1234567890	12/31/2009	Inventory	Consolidated Entity [Member]	1,000,000	USD	Thousands

A software application takes the information model description structure, the information model description rules provided, the facts that are included within the information block, and known best practices for rendering a business report that are coded into the software application in some manner and then generates a human-readable rendering of the reported information for a fragment or information block of a report.

The following is the **rendering**[111] of the inventory components disclosure that we are working with above:

---

[110] Fact Table, TO DO…

[111] Rendering, TO DO…

Reporting Entity [Aspect]	http://regulator.gov/id#1234567890	
Legal Entity [Aspect]	Consolidated Entity [Member]	
	Period [Aspect]	
Concept [Aspect]	12/31/2020	12/31/2019
Inventory, by Component [Roll Up]		
Finished Goods	600,000	600,000
Work in Progress	300,000	300,000
Raw Material	100,000	100,000
Inventory	1,000,000	1,000,000

Different software applications will provide slightly different renderings using the same XBRL-based input information.

Component: (Network and Table)		
Network	JG - Schedule - **Inventory, by Component**	
Table	Inventory, by Component [Table]	
Reporting Entity [Axis]	1234567890 http://regulator.gov/id	
Legal Entity [Axis]	Consolidated Entity [Member]	
Unit [Axis]	USD	
	Period [Axis] ▼	
Inventory, by Component [Line Items]	2010-12-31	2009-12-31
Inventory, by Component [Roll Up]		
Finished Goods	600	600
Work in Progress	300	300
Raw Material	100	100
Inventory	1,000	1,000

Here is what the information model description looks like in that software application:

Label	Report Element Class	Period	Balance	Preferred Label Role	Name
⌄ Inventory, by Component [Table]	[Table]			Standard Label	gaap:InventoryByComponentTable
⌄ Legal Entity [Axis]	[Axis]			Standard Label	frm:LegalEntityAxis
Consolidated Entity [Member]	[Member]			Standard Label	frm:ConsolidatedEntityMember
⌄ Inventory, by Component [Line Items]	[LineItems]			Standard Label	gaap:InventoryByComponentLineItems
⌄ Inventory, by Component [Roll Up]	[Abstract]			Standard Label	gaap:InventoryByComponentRollUp
Finished Goods	[Concept] Monetary	As Of	Debit	Standard Label	gaap:FinishedGoods
Work in Progress	[Concept] Monetary	As Of	Debit	Standard Label	gaap:WorkInProgress
Raw Material	[Concept] Monetary	As Of	Debit	Standard Label	gaap:RawMaterial
Inventory	[Concept] Monetary	As Of	Debit	Standard Label	gaap:Inventory

Here is what the roll up rule relations representation looks like in that software application:

Label	Report Element Class	Balance	Weight	Name
⌄ Inventory, by Component [Table]	[Table]		0	gaap:InventoryByComponentTable
⌄ Inventory	[Concept] Monetary	Debit	0	gaap:Inventory
Finished Goods	[Concept] Monetary	Debit	1	gaap:FinishedGoods
Work in Progress	[Concept] Monetary	Debit	1	gaap:WorkInProgress
Raw Material	[Concept] Monetary	Debit	1	gaap:RawMaterial

Software applications use the rule relations that describe or explain the relations to verify that reported facts are consistent with that explanation. Here is the software application interface for verifying that the reported facts are consistent with the rules that explain the relations between the facts:

Label	Rendered Value	Op	Reported Value	Calculated Value	Balance	Result	Name
∨ **Inventory, by Component [Line Items]**							gaap:InventoryByComponentLineItems
∨ **Inventory, by Component [Roll Up]**							gaap:InventoryByComponentRollUp
Finished Goods	600	+	600		Debit		gaap:FinishedGoods
Work in Progress	300	+	300		Debit		gaap:WorkInProgress
Raw Material	100	+	100		Debit		gaap:RawMaterial
Inventory	1,000		1,000	1,000	Debit	Verified	gaap:Inventory

Label	Rendered Value	Op	Reported Value	Calculated Value	Balance	Result	Name
∨ **Inventory, by Component [Line Items]**							gaap:InventoryByComponentLineItems
∨ **Inventory, by Component [Roll Up]**							gaap:InventoryByComponentRollUp
Finished Goods	600	+	600		Debit		gaap:FinishedGoods
Work in Progress	300	+	300		Debit		gaap:WorkInProgress
Raw Material	100	+	100		Debit		gaap:RawMaterial
Inventory	1,000		1,000	1,000	Debit	Verified	gaap:Inventory

Information about the properties of each report element which makes up the information model description should be accessible to the user of the financial report:

Information about the properties of each fact which is represented within the report is accessible to the user of the financial report:

This same information is provided for each and every information block that makes up a financial report. Facts could be used in multiple information blocks. The facts used in information blocks must be consistent within a information block and between the individual information blocks that make up a report.

3.10. Information Model

A financial report can be broken down into fragments. A **fragment**[112] is a set of one-to-many *information blocks* which go together for some specific purpose within a report.

For example, a balance sheet is a fragment of a financial report that is made up of two fact sets: a roll up of assets and a roll up of liabilities and equity. Fragments can be categorized into Document, Statement, Disclosure, and Schedule per some XBRL-based digital financial reporting systems.

Each fact set has a concept arrangement pattern property. A **concept arrangement pattern**[113] specifies the nature of the relationship between the concept aspect of an information model definition.

This information model patterns are identifiable by software applications[114].

- A **set**[115] is a type of concept arrangement pattern where concepts have no mathematical relations. Essentially, a set is a flat list of concepts. A synonym for set is hierarchy.

- A **roll up**[116] is a type of concept arrangement pattern which represents a basic roll up type mathematical relationship: Fact A + Fact B + Fact C = Fact D (a set of items and a total).

---

[112] Fragment, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Fragment.html

[113] Concept Arrangement Pattern, http://xbrlsite.azurewebsites.net/2019/Framework/Details/ConceptArrangementPattern.html

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

[115] Set, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Set.html

[116] Roll Up, http://xbrlsite.azurewebsites.net/2019/Framework/Details/RollUp.html

- A **roll forward**[117] is a type of concept arrangement pattern which represents a basic roll forward mathematical relation: Beginning balance (stock) + change1 (flow) + change2 (flow) + change3 (flow) = Ending balance (stock).

- An **adjustment**[118] is a type of concept arrangement pattern which represents a basic mathematical reconciliation between an originally stated value and a restated value usually due to a correction or error: Originally stated balance + adjustment1 + adjustment2 + adjustment3 = restated balance.

- A **variance**[119] is a type of concept arrangement pattern which represents a mathematical difference between two reporting scenarios: Amount (projected scenario) + Amount(variance) = Amount (actual scenario).

- An **arithmetic**[120] is a type of concept arrangement pattern which represents any arbitrary mathematical relationship between a set of numeric facts. An arithmetic is comprised of some flat set of numeric concepts and a rule that represents the mathematical relation between that set of concepts. A synonym for arithmetic is complex computation.

- A **roll forward info**[121] is a type of concept arrangement pattern which represents a non-mathematical relation of information about a roll forward.

- A **text block**[122] is a type of concept arrangement pattern which represents a non-mathematical relationship in the form of prose. A text block concept arrangement pattern is comprised of exactly one concept. There are three sub classes or type of text blocks: **Level 1 Note Text Block**[123], **Level 2 Policy Text Block**[124], and **Level 3 Disclosure Text Block**[125].

Fragments of information may have additional information referred to as a member arrangement pattern.

Each fact set has a member arrangement pattern property (no explicit pattern is a pattern). A **member arrangement pattern**[126] expresses the relations between members within an aspect other than the concept aspect (which is explained by the concept arrangement pattern).

The members of an axis might be related mathematically. **Member aggregation**[127] is a type of member arrangement pattern where the members of an axis roll up the

---

[117] Roll Forward, http://xbrlsite.azurewebsites.net/2019/Framework/Details/RollForward.html

[118] Adjustment, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Adjustment.html

[119] Variance, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Variance.html

[120] Complex Computation, http://xbrlsite.azurewebsites.net/2019/Framework/Details/ComplexComputation.html

[121] Roll Forward Info, http://xbrlsite.azurewebsites.net/2019/Framework/Details/RollForwardinfo.html

[122] Text Block, http://xbrlsite.azurewebsites.net/2019/Framework/Details/TextBlock.html

[123] Level 1 Note Text Block, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Level1NoteTextBlock.html

[124] Level 2 Policy Text Block, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Level2PolicyTextBlock.html

[125] Level 3 Disclosure Text Block, http://xbrlsite.azurewebsites.net/2019/Framework/Details/Level3DisclosureTextBlock.html

[126] Member Arrangement Pattern, http://xbrlsite.azurewebsites.net/2019/Framework/Details/MemberArrangementPattern.html

[127] Member Aggregation, http://xbrlsite.azurewebsites.net/2019/Framework/Details/MemberAggregation.html

same as the roll up concept arrangement pattern. **Member nonaggregating**[128] is a type of member arrangement pattern where the members of an axis are not related mathematically but simply are used to differentiate reported facts.

Reported facts could need additional arbitrary descriptive information. A **parenthetical explanation**[129] provides additional descriptive information about a fact. A synonym for parenthetical information is comment.

3.11. Introducing the Hypercube

We want to now introduce you to the notion of the Hypercube. A **Hypercube** is a mechanism for defining a structure in the XBRL technical syntax. Before now, we were using another mechanism for defining structures in XBRL, the **Network**. To round out the three different approaches to defining a structure there is also the approach of using a combination of a **Network and a Hypercube** to define a structure.

The first question you might have is, "Why three approaches?" That is a very good question but providing a good answer is harder. For now, don't worry about the answer to the question because creating XBRL-based reports using IFRS or US GAAP allows either approach. The "best approach" can be a matter of personal preference. Other times you will not have a choice.

Both a Hypercube and a Network are ways of connecting associations together to define a structure. Networks must always exist in XBRL-based reports. Hypercubes are optional until you need what only they can provide to structure information effectively.

Essentially, Networks and Hypercubes connect sets of associations and facts together. What we try and do is make the notion of a Network and Hypercube invisible to you and expose only the notion of the Structure itself. This is done by automatically generating the Network, Hypercube, or combination of Network + Hypercube by cleverly designing software.

But, let's forget about all that for now and look at a Hypercube.   We do this by looking at the following structure of the SFAC 6 representation:

---

[128] Member Nonaggregating, http://xbrlsite.azurewebsites.net/2019/Framework/Details/MemberNonaggregation.html
[129] Parenthetical Explanation, http://xbrlsite.azurewebsites.net/2019/Framework/Details/ParentheticalExplanation.html

Component: (Network and Table)	
Network	**02-Comprehensive Income** (http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome)
Table	Comprehensive Income Statement [Hypercube]

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

Reporting Entity [Axis]	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)

	Period [Axis]
Comprehensive Income Statement [Line Items]	2020-01-01 - 2020-12-31
Comprehensive Income [Roll Up]	
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Comprehensive Income	3,000

Notice the "Comprehensive Income Statement [Hypercube]" in the representation above. In this case whether the Hypercube does or does not exist changes nothing about the meaning of the information conveyed by the XBRL-based information provided.

We are going to cover Hypercubes more in a bit. But first, we need to Segway by looking at a pathological structure in order to better understand structures.

3.12. Pathological Structure

An excellent way to understand a good structure is to look at bad structures and compare them with good structures.

What you see below is four logical structures that have been represented as one physical structure within an XBRL-based report[130]:

---

[130] Pathological structure, http://xbrlsite.azurewebsites.net/2020/introduction/pathological/evidence-package/

Structure [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Comprehensive Income	3,000
Net Cash Flow Operating Activities	1,500
Net Cash Flow Investing Activities	1,000
Net Cash Flow Financing Activities	1,000
Net Cash Flow	3,500
Assets, Beginning Balance	0
Net Cash Flow	3,500
Assets, Ending Balance	3,500
Revenues	7,000
Comprehensive Income	3,000
Distributions to Owners	500

The best way to understand why the pathological representation is not elegant, let's look at a minor improvement to that pathological example which separates the structures better[131]:

Reporting Entity [Axis]	GH259400TOMPUOLS65II http:/
	Period [Axis]
Implied [Line Items]	2020-01-01/2020-12-31
Block 1 [Abstract]	
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Comprehensive Income	3,000
Block 2 [Abstract]	
Net Cash Flow Operating Activities	1,500
Net Cash Flow Investing Activities	1,000
Net Cash Flow Financing Activities	1,000
Net Cash Flow	3,500
Block 3 [Abstract]	
Assets, Beginning Balance	0
Net Cash Flow	3,500
Assets, Ending Balance	3,500
Block 4 [Abstract]	
Revenues	7,000
Comprehensive Income	3,000
Distributions to Owners	500

---

[131] Improved pathological representation, http://xbrlsite.azurewebsites.net/2020/introduction/patho-better/evidence-package/

While the above is a bit easier to read and understand because of the addition of four [Abstract] report elements which are used to separate the four structures; however, there is still room for improvement.

An even better representation of exactly the same information makes the information much clearer[132]:

Income Statement [Roll Up]	Period [Axis]
	2020-01-01 - 2020-12-31
Income Statement [Roll Up]	
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Comprehensive Income	3,000

Cash Flow Statement [Abstract]	Period [Axis]
	2020-01-01 - 2020-12-31
Cash Flow Statement [Abstract]	
Net Cash Flow [Roll Up]	
Net Cash Flow Operating Activities	1,500
Net Cash Flow Investing Activities	1,000
Net Cash Flow Financing Activities	1,000
Net Cash Flow	3,500
Assets [Roll Forward]	
Assets, Beginning Balance	0
Net Cash Flow	3,500
Assets, Ending Balance	3,500

Financial Highlights [Set]	Period [Axis]
	2020-01-01 - 2020-12-31
Financial Highlights [Set]	
Revenues	7,000
Comprehensive Income	3,000
Distributions to Owners	500

Notice how the four structures (i.e. income statement which is a roll up, net cash flow roll up and assets roll forward which makes up the cash flow statement, and the unrelated set of facts which make up the financial highlights set) are significantly more understandable.

---

[132] Best pathological representation, http://xbrlsite.azurewebsites.net/2020/introduction/patho-best/evidence-package/

While it is true that all three versions of this same example convey exactly the same meaning and all drive the same Inline XBRL representation (see next page); it is likewise true that all things considered the third representation that you see above is more explicit and tends to be easier to understand. Here is the same information represented using Inline XBRL[133]:

Statement of Comprehensive Income

ABC Company, Inc.

(See accompanying notes to the financial statements.)

(in US Dollars)	For period ended December 31, 2020
Comprehensive income:	
Revenue	$7,000
Expenses	(3,000)
Gains	1,000
Losses	(2,000)
Comprehensive income	$3,000

Cash Flow Statement

ABC Company, Inc.

(See accompanying notes to the financial statements.)

(in US Dollars)	For period ended December 31, 2020
Net cash flow:	
Net cash flow from operating activities	$1,500
Net cash flow from investing activities	1,000
Net cash flow from financing activities	1,000
Net cash flow	3,500
Assets, beginning balance	0
Assets, ending balance	$3,500

Financial Highlights

ABC Company, Inc.

(in US Dollars)	For period ended December 31, 2020
Financial highlights:	
Revenues	$7,000
Comprehensive income	3,000
Distributions to owners	500

---

[133] Inline XBRL of pathological representation,
http://xbrlsite.azurewebsites.net/2020/introduction/pathological/instance-RENDERED.html

Notice how the third representation is the most consistent with the Inline XBRL representation and note how a better model tends to yield a better automated rendering of information without the need for taking the time to create the Inline XBRL.

4. Proof of Logical Conceptualization

OK, so we explained the logical conceptualization of a financial report. How do we know the logical conceptualization works and exactly how to represent that conceptualization using the XBRL technical syntax? Well, we build a simple proof that exercises 100% of the logical conceptualization to test it and show how to construct it to prove that it does work correctly.

We are going to do this in two steps. In the first step we are going to build each piece of the Proof[134] to see that each piece works as expected and in the second step we are going to put all of the pieces together into one XBRL-based report Proof[135] to be sure all the pieces work together as would be expected.

We are not going to walk you through building each piece because this is just a mechanical exercise. We will provide you with a complete example of each piece so that you can see every detail. We strongly suggest that you create each piece and compare what you created with the example we provided.

We are going to start with the easiest pieces of the Proof and with pieces with which you have already been exposed to in the accounting equation, SFAC 6, common elements of financial report, and Hello World examples. Where something new is introduced, we will take the time to fill you in on those new details.

For each piece we will show you the model structure and point you to all of the details. Included in the details is a version of the free open source Microsoft Access database application with the completed piece. Also included is an XBRL Cloud Evidence Package you can have a look at.

Let's get started with the most basic piece, a simple set of financial highlights.

4.1. Set[136]

The financial highlights of the Proof is an example of a set information model. A set is simply some set of facts. Here we have a set of three facts:

Financial Highlights [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Financial Highlights [Set]	
Revenues	7,000
Comprehensive Income	3,000
Distributions to Owners	500

---

134 *Information Models*, http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/index.html

135 *Proof Baseline*, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/index.html

136 Set, http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/set/index.html

Not much else to point out here.

4.2. Roll Up[137]

The income statement (alternative) is a basic example of a roll up information model. Here you again see three facts. The difference between the set and the roll up is that the roll up has a rule that says the first two facts "roll up" to the third fact which is the total of the roll up:

Income Statement [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Net Income [Roll Up]	
Income from Normal Activities of Entity	2,000
Income from Peripheral or Incidental Transactions of Entity	1,000
Net Income	3,000

For a roll up there is always a rule that specifies the mathematical relations between the facts which is expressed using either XBRL calculation relations or XBRL formulas.

4.3. Roll Forward[138]

The statement of changes in equity is a basic example of a roll forward information model. Again, you have some set of facts and a mathematical relation; except the mathematical relation of a roll forward is different from the mathematical relation of a roll up. A roll forward reconciles a beginning balance to an ending balance via some set of changes. You see this below:

Changes in Equity [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Changes in Equity [Roll Forward]	
Equity, Beginning Balance	0
Comprehensive Income	3,000
Investments by Owners	1,000
(Distributions to Owners)	(500)
Equity, Ending Balance	3,500

XBRL calculation relations are only for roll ups, so we need to use XBRL Formula to represent the roll forward mathematical rule for this piece of the report.

---

[137] Roll Up, http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/rollup/index.html
[138] Roll Forward,
http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/rollforward/index.html

4.4. Adjustment[139] (a.k.a. Restatement)

A prior period adjustment is an example of the adjustment information model. Once again, we have some set of facts and a mathematical relation between the set of facts. An adjustment reconciles an originally stated balance to a restated balance via some set of adjustments. You see this below:

Prior Period Errors [Line Items]	Report Date [Axis]	Period [Axis] 2019-12-31
Prior Period Errors [Adjustment]		
Equity, Origionally Stated	Prior Report [Member]	2,000
Changes in Accounting Policy	Current Report [Member]	(1,500)
Correction of an Error	Current Report [Member]	(500)
Equity, Restated	Current Report [Member]	0

XBRL calculation relations are only for roll ups, so once again we need to use XBRL Formula to represent the adjustment mathematical rule for this piece of the report.

4.5. Adjustment (Alternative)[140]

There is an alternative approach that is also used to sometimes represent a prior period adjustment. While the syntax used to represent the adjustment (alternative) we show here and the adjustment we show above; the semantics or meaning of the information is 100% the same for the two different approaches.

Here the same prior period adjustment is represented using XBRL Dimensions to differentiate each fact rather than using concepts. You see this below:

Prior Period Errors [Line Items]	Period [Axis] 2019-12-31			
	Restatement [Axis]			
	Changes in Accounting Policy [Member]	Correction of Accounting Errors [Member]	Previously Stated [Member]	Restated [Member]
Prior Period Errors [Adjustment]				
Equity	(1,500)	(500)	2,000	0

Again, XBRL Formula must be used to represent the adjustment mathematical rule for this alternative piece of the report.

4.6. Variance[141] (a.k.a. Difference)

The variance analysis is a basic example of a variance information model. Once again, you have some set of facts and a mathematical relation. Here the mathematical relation of the variance is the difference between a budgeted amount and an actual amount per some variance. You see this below:

---

139 Adjustment,
http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/adjustment/index.html
140 Adjustment (Alternative),
http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/adjustment2/index.html
141 Variance, http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/variance/index.html

Variance Analysis [Line Items]	Period [Axis]		
	2020-01-01 - 2020-12-31		
	Scenario [Axis]		
	Budgeted [Member]	Variance [Member]	Actual [Member]
Variance Analysis [Roll Up]			
Revenues	6,000	1,000	7,000
(Expenses)	(2,000)	(1,000)	(3,000)
Gains	750	250	1,000
(Losses)	(1,000)	(1,000)	(2,000)
Net Income	3,750	(750)	3,000

Again, XBRL Formula must be used to represent the variance mathematical rule for this piece of the report. XBRL calculations or XBRL formula can be used to represent the roll up mathematical relations.

There is something else going on here that is worth pointing out. In this report piece you actually have two different information model patterns that work together to represent the information. In addition to the variance, you see a Roll Up information model. If the line item "Net Income" were removed; then the additional information model would be a Set information model. This could just as well be a Roll Forward or an Adjustment information model. The point here is that information models can work together with each other.

4.7. Roll Forward Info[142]

The weighted average grant date fair value piece is a basic example of a roll forward info information model. There is one subtle difference between a roll forward and a roll forward info model. A roll forward has a mathematical rule and a roll forward looks like it has a mathematical rule per the formatting; but there really is no mathematical relation between the set of facts. You see this below:

Weighted Average Grant Date Fair Value [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Weighted Average Grant Date Fair Value [Roll Forward Info]	
Nonvested Fair Value, Beginning Balance	32.72
Granted	41.51
Vested	30.92
Forfeited	35.93
Nonvested Fair Value, Ending Balance	36.92

Nothing else to point out.

---

[142] Roll Forward Info,
http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/rollforwardinfo/index.html

4.8. Text Block[143]

The policies piece is a basic example of a text block information model. A text block is simply one fact that contains prose of any sort. There are no mathematical relations or other facts to be associated with. Each individual text block is always an individual information model.

Policies [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Revenue Recognition Policy [Text Block]	Sed mauris. Nulla facilisi. Fusce tristique posuere ipsum. Nulla facilisi. Aliquam viverra risus vitae ante. Sed rhoncus mi in wisi. Nullam nibh dui, molestie vitae, imperdiet non, ornare at, elit. • Suspendisse accumsan, arcu vel ornare interdum, magna tellus porta mauris, in porta mi lacus sodales felis. • Phasellus eleifend, diam vitae dapibus pulvinar, erat ligula auctor dui, eget congue justo lorem hendrerit tellus. • Fusce gravida, ligula a placerat placerat, leo erat euismod lectus, et lacinia justo libero non pede. Fusce gravida, ligula a placerat placerat, leo erat euismod lectus, et lacinia justo libero non pede. Vivamus ac velit vel magna nonummy pretium. 1. Etiam ut augue 2. Aliquam erat volutpat

Note that the Proof has three text blocks and we are only modeling one text block for you here because the other two would be the same as the one shown above; just different concepts.

4.9. Arithmetic[144]

The accounting equation is a basic example of an arithmetic information model. An arithmetic information model is simply some set of facts and some mathematical rule that describes how the facts are mathematically related to one another. You see this below:

Balance Sheet [Line Items]	Period [Axis]	
	2020-12-31	2019-12-31
Balance Sheet [Arithmetic]		
Assets	3,500	0
Liabilities	0	0
Equity	3,500	0

XBRL Formula must be used to represent the arithmetic mathematical rule and any rule that can be represented using XBRL Formula is a valid rule that can be used to represent this information model.

Here, the rule is "Assets = Liabilities + Equity"; the accounting equation.

4.10. Member Aggregation[145]

The segment revenues piece is a basic example of a member aggregation information model. A member aggregation is some set of facts that have one total and some set of members that aggregate to that total. Semantically, a member aggregation is logically equivalent to a roll up. You see this below:

---

[143] Text Block, http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/textblock/index.html

[144] Arithmetic, http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/arithmetic/index.html

[145] Member aggregation, http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/memberaggregation/index.html

Segment Revenues [Line Items]	Period [Axis]			
	2020-01-01 - 2020-12-31			
	Segments [Axis]			
	Segment Alpha [Member]	Segment Bravo [Member]	Segment Charlie [Member]	All Segments [Member]
Sgement Revenues [Set]				
Revenues	1,000	4,000	2,000	7,000

XBRL Formula must be used to represent the member aggregation mathematical rule for this piece of the report.

4.11. Member Nonaggregation[146]

The weighted average grant date fair value piece of the report, with one modification, is a basic example of a member nonaggregation information model. As we mentioned before, information models can be used together. Here, roll forward infos are provided for three different plans and no total is shown and therefore we have a set of members with no aggregation. You see this below:

Weighted Average Grant Date Fair Value [Line Items]	Period [Axis]		
	2020-01-01 - 2020-12-31		
	Plan [Dimension]		
	Plan One [Member]	Plan Two [Member]	Plan Three [Member]
Weighted Average Grant Date Fair Value [Roll Forward Info]			
Nonvested Fair Value, Beginning Balance	1.72	2.72	3.72
Granted	1.51	2.51	3.51
Vested	1.92	2.92	3.92
Forfeited	1.93	2.93	3.93
Nonvested Fair Value, Ending Balance	1.92	2.92	3.92

Again, there is no mathematical rule because there is no mathematical relation between any of the facts shown in this piece.

4.12. Component[147]

The balance sheet piece of the Proof report is a basic example of a component information model. A component is simply some set of information models that make up a complex information model.

The balance sheet is comprised of two other information models: (1) an assets roll up and (2) a liabilities and equity roll up. Each of the information models in the component are controlled by the rules for that specific information model.

In addition, it could be the case that additional rules come into play because you have more than one fact set. Here the total of the assets roll up must equal the total of the

---

[146] Member Nonaggregation,
http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/membernonaggregation/index.html
[147] Component,
http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/component/index.html

liabilities and equity roll up because a fundamental rule in accounting is that "Assets = Liabilities and Equity" (the double entry accounting model). You see this below:

Balance Sheet [Line Items]	Period [Axis]	
	2020-12-31	2019-12-31
Assets [Roll Up]		
Current Assets	500	0
Noncurrent Assets	3,000	0
Assets	3,500 [1]	0 [1]
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	3,000	0
Equity Attributable to Noncontrolling Interests	500	0
Equity	3,500	0
Liabilities and Equity	3,500	0

4.13. Component (Second Example)[148]

The cash flow statement piece of the Proof report is another basic example of a component information model. You see this below:

Cash Flow [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Net Cash Flow [Roll Up]	
Net Cash Flow Operating Activities	1,500
Net Cash Flow Investing Activities	1,000
Net Cash Flow Financing Activities	1,000
Net Cash Flow	3,500
Assets Roll Forward	
Assets, Beginning Balance	0 [1]
Net Cash Flow	3,500
Assets, Ending Balance	3,500 [1]

---

[148] Component 2,
http://xbrlsite.azurewebsites.net/2021/essentials/InformationModel/component2/index.html

What is different here and worth showing is that components can be made up of different information models. Here you see a roll up of net cash flow and a roll forward of assets (because we don't have a Cash and Cash Equivalents line item on the balance sheet for this PROOF report).

4.14. Proof Baseline

The Proof Baseline is a comparison of the logic found in three financial reporting schemes: US GAAP, IFRS, UK GAAP, and AU GAAP[149]. The PROOF is then reconciled to those four financial reporting schemes.

4.15. Proof

The Proof[150] representation pulls each of the individual information model representations into one combined report to be sure (a) individual information models work effectively and (b) individual information models interact correctly to other information models to represent a complete report. The interrelation between report pieces can be visualized as such[151]:
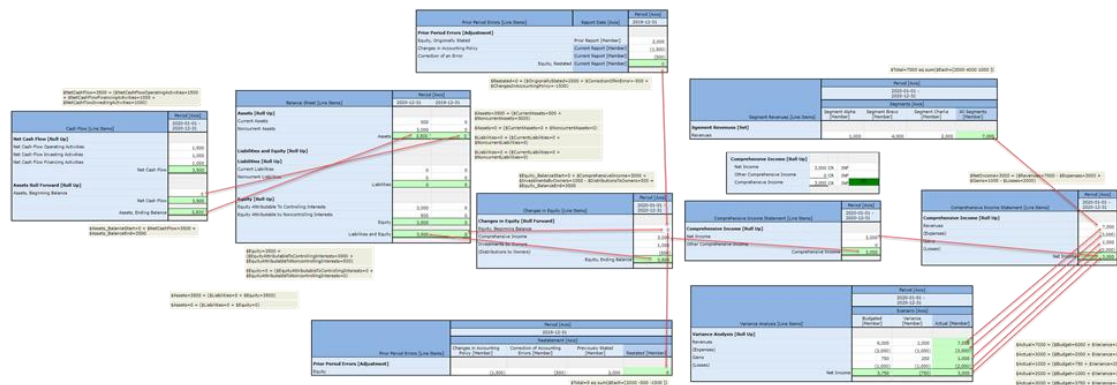
The combined Proof Baseline report for which information is represented in 14 networks which have 14 hypercubes which contain 18 unique "blocks" of information[152]. Everything that you would ever run across in any XBRL-based financial report using US GAAP, UK GAAP, IFRS, or as far as I can tell any other financial reporting scheme is contained within the Proof Baseline representation.

In fact, I have reconciled a version of the Proof Baseline to US GAAP, UK GAAP, IFRS, Australian IFRS to compare and contrast the Proof Baseline with other important financial reporting schemes[153]. One would expect that each financial reporting scheme would represent the same

---

[149] Proof Baseline, http://xbrl.squarespace.com/journal/2020/12/19/mechanism-for-understanding-xbrl-based-financial-report-sema.html

[150] Proof, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/index.html

[151] Proof, Mathematical relations, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/ProofMathematicalComputations.jpg

[152] Proof, XBRL Cloud Evidence Package, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/evidence-package/contents/index.html#Rendering-BalanceSheet-proof_BalanceSheetHypercube.html

[153] Mechanism for Understanding XBRL-based Financial Report Semantics, http://xbrl.squarespace.com/journal/2020/12/19/mechanism-for-understanding-xbrl-based-financial-report-sema.html

logical associations in the same manner. Or saying this another way, why would US GAAP represent a roll up one way and IFRS represent a roll up in some other way? A roll up is a roll up no matter what the financial reporting scheme.

I would encourage you to take the time to create the Proof Baseline representation in an XBRL-based financial report creation tool of your choice to fully understand the important information provided by this Proof Baseline representation.

5.   Hypercubes, Networks, and Structures

There are a few additional nuances and subtleties we need to point out related to hypercubes, networks, and the structures they represent. First, there are two reasons you create a new structure: (1) because you *have to* and (2) because you *want to*.

5.1. Networks

There are two important aspects to keep in mind when you think about XBRL networks. The first you have already been exposed to which is to **separate the fragments** which make up a report. As we said above, you separate things because you *have to* or because you *want to*. Sometimes it does not matter logically if two fragments are combined or if those same two fragments are separated into different networks, separating the fragments simply might be more elegant when you look at the XBRL-based digital financial report than when the two fragments are combined.

But the other thing an XBRL network does is combined information across the different technical mechanisms for creating associations. An example will make this very easy to see and understand.

Below you see one fragment of a report represented within one network and you see the three common types of associations that are represented in XBRL-based reports: XBRL presentation relations, XBRL calculation relations, and XBRL definition relations:
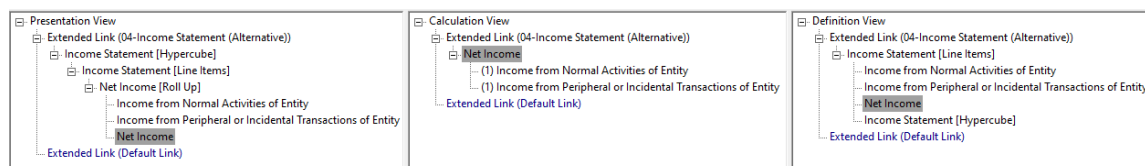
How do you, or more importantly a software application, understand that the three sets of information go together? The answer is that they have the same network identifier. Each of the sets of XBRL presentation relations, XBRL calculation relations, and XBRL definition relations use the same network identifier to indicate that the information in each set of associations goes together. It is also the case that XBRL Formulas uses networks to indicate which network an XBRL formula resource goes with.

Finally, XBRL has a default network that can be used to indicate that information is global. For example, XBRL labels and XBRL references use the default role to identify information because labels and references tend to always relate globally throughout a report as opposed to be used only for one specific network.

Saying this another way; while networks are used to separate report fragments (i.e. vertical structures) the same networks are use to combine XBRL technical fragments (i.e. horizontal structures) so that users of the information understands that the technical fragments go together.

5.2. Structures

You *have to* create a new structure when a conflict occurs when you do not. For example, in the Proof Baseline representation there are two different roll up alternatives provided for an Income Statement which each has a different roll up of Net Income we provide here:

Alternative 1[154]:

Comprehensive Income Statement [Line Items]	Period [Axis]
	2020-01-01 - 2020-12-31
Comprehensive Income [Roll Up]	
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Net Income	3,000

---

[154] Income Statement Roll Up 1, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/evidence-package/contents/index.html#Rendering-ComprehensiveIncome-proof_ComprehensiveIncomeStatementHypercube.html

Alternative 2[155]:

	Period [Axis]
Comprehensive Income Statement [Line Items]	2020-01-01 - 2020-12-31
Comprehensive Income [Roll Up]	
Income from Normal Activities of Entity	2,000
Income from Peripheral or Incidental Transactions of Entity	1,000
Net Income	3,000

It is literally impossible to represent both alternatives using the same physical structure, so two different structures are required to be created. In this case two networks are used to keep the two different mathematical associations from "logically colliding" with one another. There is no way around this.

But sometimes you just want to create another structure. You can do that also. For example, you could represent the balance sheet and income statement within the same structure because each of those disclosures use completely separate terms, associations, and rules so you would not get a logical conflict if you did put those two structures together into the same network or even within the same hypercube. Or, you could make the choice to use two separate structures. In this circumstance, either approach works fine logically because there are no logical conflicts. However, aesthetically it is good practice to separate these into two different networks.

5.3. Hypercubes

And so, when you do create a new structure, we pointed out that there are three alternatives for doing so when it comes to whether you do or do not use hypercubes and how you name those hypercubes. Here we show each of those three approaches. We will contrast the income statement of the three approaches to help you differentiate the approaches.

Approach 1 (No Hypercubes)[156]: SFAC 6 Very Basic representation

In this first approach, note that neither the balance sheet, either income statement, nor changes in equity have hypercubes. In each case the Table (a.k.a. Hypercube) field says "(Implied)" which means that no explicit hypercube has been defined and therefore an imaginary hypercube that does not really exist is implied. Each network has a unique identifier and can be used to distinguish one structure from another.

---

[155] Income Statement (alternative) Roll Up 2, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/evidence-package/contents/index.html#Rendering-ComprehensiveIncome2-proof_ComprehensiveIncomeStatementHypercube.html
[156] No hypercubes, SFAC 6 Very Basic, http://xbrlsite.azurewebsites.net/2021/essentials/Hypercubes/sfac6-NoHypercube/index.html

Component: (Network and Table)	
Network	**02-Comprehensive Income** (http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome)
Table	(Implied)

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

Reporting Entity [Axis]	GH259400TOMPUOLS65II

	Period [Axis]
Comprehensive Income [Roll Up]	2020-01-01 - 2020-12-31
Comprehensive Income [Roll Up]	
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Comprehensive Income	3,000

However, consider the following. What if you combined the Comprehensive Income [Roll Up] and the Balance Sheet [Set] into the same network. Could you use the Network to differentiate one structure from the other when they share the same network? The answer is no; you would need to provide additional information.

Approach 2 (Unique Hypercubes)[157]: SFAC 6 Introducing Hypercubes representation

In this second approach, notice that every structure provides a hypercube; the balance sheet, each income statement, and the changes in equity. But notice one important detail. Both of the income statements use the same hypercube name, "Comprehensive Income Statement [Hypercube]".

So, while each structure does contain a hypercube, because the hypercubes are not uniquely named, the hypercube cannot be used to distinguish or identify structures. While you can distinguish the balance sheet and statement of changes in equity; you cannot do the same with the two income statements that use the same hypercube.

Or, saying this another way; if each of the hypercubes did have a unique name, then the hypercube alone could be used to differentiate each structure.

---

[157] SFAC 6 Introducing Hypercubes, http://xbrlsite.azurewebsites.net/2021/essentials/Hypercubes/sfac6-UniqueHypercube/index.html

Component: (Network and Table)	
Network	**02-Comprehensive Income** (http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome)
Table	Comprehensive Income Statement [Hypercube]

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

Reporting Entity [Axis]	GH259400TOMPUOLS65II

	Period [Axis]
Comprehensive Income Statement [Line Items]	2020-01-01 - 2020-12-31
Comprehensive Income [Roll Up]	
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Comprehensive Income	3,000

Approach 3 (Same Hypercube)[158]: SFAC 6 Alternative Hypercube Approach representation

In this alternative representation using hypercubes, the approach where one hypercube named "Hypercube [Hypercube]" is used to represent the balance sheet, each of the two income statements, and the changes in equity. See the first income statement here and look at the line below to see each of the others. Further note that the Line Items term, "Hypercube [Line Items]", is used in the representation of each hypercube.

Component: (Network and Table)	
Network	**02-Comprehensive Income** (http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome)
Table	Hypercube [Hypercube]

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

Reporting Entity [Axis]	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)

	Period [Axis]
Hypercube [Line Items]	2020-01-01 - 2020-12-31
Comprehensive Income [Roll Up]	
Revenues	7,000
(Expenses)	(3,000)
Gains	1,000
(Losses)	(2,000)
Comprehensive Income	3,000

What does sharing hypercubes and the same line items term between structures achieve? Well, first you save having to create a new hypercube and the line items terms; you simple have to

---

[158] SFAC 6 Alternative Hypercube Approach, http://xbrlsite.azurewebsites.net/2021/essentials/Hypercubes/sfac6-SameHypercube/index.html

create them once and use them many times which reduces the number of terms you must define.

Second, it forces one to use the network to physically identify the structure. Because each hypercube has the same name, it is useless in the identification of which structure you are referring to.

Third, because you use the same hypercube name the creator of XBRL taxonomies is forced to only use one hypercube per network. Why? If you only have one hypercube to use and you put them in the same network, then conflicts will occur when associations between terms are defined within a network for hypercubes that have the same name. The vast majority of XBRL-based reports already have only one hypercube her network; this forces every report to use this approach.

Fourth, when you do want to extract information from a report and you want to leverage a hypercube for doing so but each hypercube has the same name plus network identifiers are of no help because both the SEC and ESMA require reporting entities to define and use their own network identifiers (as contrast to each company having to use a network identifier defined by the FASB or IASCF or SEC or ESMA) how can you possibly extract information from a report for a specific structure?

Well, that is an excellent question! The answer to that important question is to use what is known as prototype theory[159] which is beyond the scope of this introduction. In a nutshell, you use the parts of a structure to identify the structure as contrast to a unique name that identifies each structure.

5.4. Component

A component is a combination of a Network PLUS one Hypercube. A Hypercube might be explicitly provided or implied. Fundamentally, all information represented in an XBRL-based report exists within a Hypercube; be that Hypercube explicit or implied.

5.5. Fragment

A fragment is an arbitrary portion of a financial report.

5.6. Block

A "block" or "information block" is a logical fragment of a financial report. Blocks are not arbitrary; they are driven by logic.

5.7. Disclosure

A disclosure is some set of information blocks that go together to provide some financial disclosure.

---

[159] Understanding Prototype Theory, http://xbrl.squarespace.com/journal/2013/12/21/understanding-prototype-theory.html

5.8. Best Approach for Representing Structures

And so, this begs the question: What is the best approach for representing and working with information structures? Using Networks, Hypercubes, Components, Blocks, Disclosures, or arbitrary Fragments? Well, the answer to the question is not that simple and depends on your perspective. What if software could provide you with any of these views of the report model and report information?
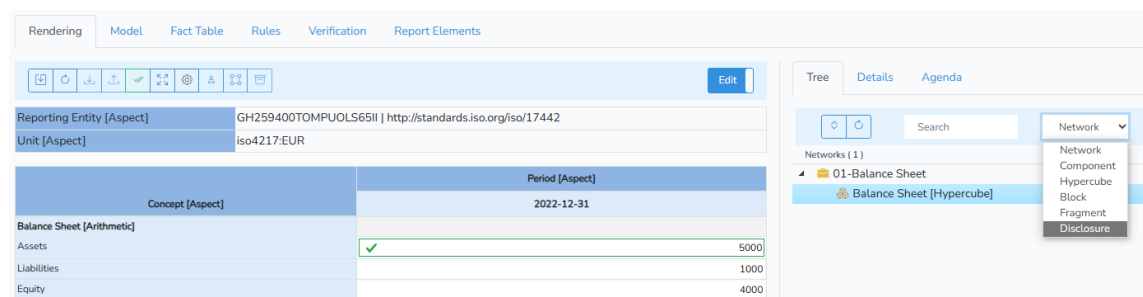
In order to define a proper XBRL taxonomy for, say, US GAAP or IFRS; you would have to know 100% of the hypercubes in advance in order to use a hypercube to identify each possible disclosure structure. While that is possible, it is challenging.

Today, most quality creators of XBRL-based reports for the SEC are using hypercubes to represent all information in a report. For example, the Microsoft 10-K[160] has 128 Networks, 128 hypercubes (a.k.a. Tables), and 194 information blocks. Every structure is represented within a hypercube, but the hypercubes are not uniquely named (i.e. that hypercube us-gaap:StatementTable is used numerous times to identify numerous structures). Therefore, hypercubes are useless for extracting information.

Internally within the enterprise, I would speculate that people would find it easier to deal with unique hypercubes for each unique disclosure.

6. Intermediate

Fundamentally, XBRL-based digital financial reports are a medium for exchanging information[161] which is important in today's digital world and which can supercharge what artificial intelligence can achieve for professional accountants.

Accounting existed before writing. Between 5,000 and 10,000 years ago farmers in Mesopotamia, where agriculture was born, used physical objects to count crops and animals[162]. The distinction between types of crops or animals was made by using different types and shapes of objects. These farmers began documenting information using clay tablets in the earliest form of human writing ever discovered called Cuneiform. They partitioned their clay tablet into rows, columns, and cells. Then, in

---

[160] Microsoft XBRL-based Report Analysis, http://xbrl.squarespace.com/journal/2020/4/13/microsoft-xbrl-based-report-analysis.html

[161] *Understanding that XBRL is a Knowledge Media*, http://xbrl.squarespace.com/journal/2017/1/16/understanding-that-xbrl-is-a-knowledge-media.html

[162] Denise Schmandt-Bessersat, *On the origins of writing*, YouTube.com, https://www.youtube.com/watch?v=kidWY-pJFb0

about 3200 BC, around 5,000 years ago, the first spreadsheet was invented. Clay tablets gave way to papyrus, then paper. In the late 20th century, paper began being replaced by electronic versions of paper such as PDF files, word processing documents, HTML files, and electronic spreadsheets.

Structured formats such as XBRL take this transition one step further making the information contained within those documents and making that information not only readable but understandable and interpretable by computer-based processes. This is an important feature in the information age as the complexity and volume of information increases and manual processes are overwhelmed.

Ultimately, software applications will be able to assist professional accountants in the repetitive, monotonous, routine, mechanical, mundane, rote processes and tasks that they perform much like a calculator augments the ability of a professional accountant to do math.

How exactly can this be made to work? Magic? There is no magic involved, just sound engineering and attention to detail. While no magic is involved the resulting functionality will seem quite magical. So how exactly do we get computers to perform this work?

6.1. Getting Computers to Effectively Perform Work

Per Wikipedia, a **computation**[163] is defined as:

> "A computation is any type of calculation that includes both arithmetical and non-arithmetical steps and which follows a well-defined model (e.g. an algorithm)."

The key takeaway here is that by computation we don't just mean math. Computation means steps that can be followed by a computer. Per Wikipedia (paraphrasing), **algorithm[164]** is defined as:

> "An algorithm is a finite sequence of well-defined instructions, typically to solve a class of problems or to perform a computation. Algorithms are always unambiguous and are used as *specifications* for performing calculations, data processing, automated reasoning, and other tasks."

Computational logic[165] is the use of logic to perform or reason about computation. Logic is used to define the steps and tasks. Logic is a set of principles that forms a framework for correct reasoning.

Humans are very capable of carrying out steps and performing tasks. Mechanical devices can also be created to carry out steps and perform tasks, for example a vending machine is such a device. A calculator is also such a device. Computers via the software programs they run can likewise carry out steps and perform tasks.

Can computers perform all work?[166] The answer is no. For example, computers cannot exercise professional judgment. But computers can help out with many routine,

---

[163] Wikipedia, *Computation*, https://en.wikipedia.org/wiki/Computation

[164] Wikipedia, *Algorithm*, https://en.wikipedia.org/wiki/Algorithm

[165] Wikipedia, *Computational Logic*, https://en.wikipedia.org/wiki/Computational_logic

[166] Artificial Intelligence and Knowledge Engineering Basics in a Nutshell, http://xbrlsite.azurewebsites.net/2019/Library/KnowledgeEngineeringInNutShell.pdf

repetitious, monotonous, mechanical, boring, grueling tasks and processes of accounting, reporting, auditing, and analysis.

So again, how do you get computers to perform work? 'Hope' is not a sound engineering principle. To get computers to do work effectively one needs "know how". Know how is a type of practical knowledge. A **best practice** is a method or technique that has been generally accepted as superior to any other known alternatives because it produces results that are superior to those achieved by other means or because it has become a standard way of doing things. This information provides an introduction to a tested, proven, best practices-based method for creating XBRL-based financial reports[167].

Key to getting computers to perform work effectively is controlling processes. Key to controlling computer-based processes is using computer understandable rules.

6.2. Control + Rules = Effective Automation (High Quality)

If a process cannot be controlled then the process simply cannot repeatedly and reliably output high-quality. If process output is not high-quality, automation cannot possibly be effective.

So, control of a process is necessary in order for the process to be effective. How do you control a process? You control a process using rules. Manual processes are controlled by rules that are read by humans. Automated processes are controlled by rules that are readable by both machines (i.e. to execute the process) and humans (i.e. to make sure the rules are right).

Who creates these machine-readable rules that are used to control processes that yield effective automation? Accountants must create these rules because the rules tend to be accounting oriented. Technical rules tend to relate to rule syntax and such technical rules can be hidden from business professionals. What is left is the business logic and accounting rules that are used to control information and control process workflow. As such, the creation of machine-readable rules must be "self-service". Business professionals must be empowered to create, adjust, maintain, and otherwise manage the rules that are used to control and therefor effectively automate processes.

We covered this in the introduction, but we will repeat our explanation of what a logical theory is because it introduces a bit of important jargon we need to communicate effectively about this topic.

6.3. Accounting Equation, a Simple Financial Report

The double-entry accounting model and the accounting equation provide the very foundation of financial reporting. There are many different sources for the accounting equation, but I will use Wikipedia's version[168], of the accounting equation: **Assets = Liabilities + Equity**

As Wikipedia points out, that fundamental accounting equation, also called the balance sheet equation, represents the relationship between the line items assets, liabilities, and equity of an economic entity. It is the foundation for the double-entry bookkeeping system. For each transaction, the total debits equal the total credits.

---

[167] *Understanding Method (Abridged)*,
http://xbrlsite.azurewebsites.net/2020/Library/UnderstandingMethod_Abridged.pdf
[168] Wikipedia, Accounting Equation, https://en.wikipedia.org/wiki/Accounting_equation

We can leverage this very simple equation to understand the fundamentals of XBRL-based financial reporting and learn important subtleties and nuances of working with this new digital report format.

We represented this accounting equation in the form of a model which looked something like the following in our gentle introduction to XBRL-based financial reports[169]:

Label	Object Class	Period Type	Balance	Report ElementName
Balance Sheet [Arithmetic]	Abstract			ae:BalanceSheetArithmetic
Assets	Concept (Monetary)	As Of	Debit	ae:Assets
Liabilities	Concept (Monetary)	As Of	Credit	ae:Liabilities
Equity	Concept (Monetary)	As Of	Credit	ae:Equity

As every accountant knows the accounting equation can be represented in another way to make it more appropriate for not for profit entities. That alternative equation is "**Net Assets = Assets - Liabilities**" and when represented in XBRL might look something like the following:

Label	Object Class	Period Type	Balance	Report ElementName
Net Assets [Roll Up]	Abstract			ae:NetAssetsRollUp
Assets	Concept (Monetary)	As Of	Debit	ae:Assets
Liabilities	Concept (Monetary)	As Of	Credit	ae:Liabilities
Net Assets	Concept (Monetary)	As Of	Debit	ae:NetAssets

So, what if you wanted to combine the first representation and the second representation together, how can you achieve that? You have to come up with some term that describes the first set of things and the second set of things.

That term is what we call a **structure**. As we defined the notion of a structure; it is a set of statements that describe the structure.

Here we combine the first set and the second set together and create the structures "Balance sheet" and "Statement of Net Assets" to differentiate the first structure from the second structure[170] using a name for the structure:

Line	Label	Object Class	Period Type	Balance	Report ElementName
1	01-Balance Sheet	Network			http://www.xbrlsite.com/ae/role/BalanceSheet
2	Balance Sheet [Arithmetic]	Abstract			ae:BalanceSheetArithmetic
3	Assets	Concept (Monetary)	As Of	Debit	ae:Assets
4	Liabilities	Concept (Monetary)	As Of	Credit	ae:Liabilities
5	Equity	Concept (Monetary)	As Of	Credit	ae:Equity
6	02-Statement of Net Assets	Network			http://www.xbrlsite.com/ae/role/StatementOfNetAssets
7	Net Assets [Roll Up]	Abstract			ae:NetAssetsRollUp
8	Assets	Concept (Monetary)	As Of	Debit	ae:Assets
9	Liabilities	Concept (Monetary)	As Of	Credit	ae:Liabilities
10	Net Assets	Concept (Monetary)	As Of	Debit	ae:NetAssets

The statements that describe each of the terms used by each structure and the associations between those terms describe the structure. Do the structures have names? Can they be identified by someone who is looking for each of the structures in a report?

Note that for XBRL-based reports submitted to the SEC and to the ESMA, the networks that are used to define the structures as is being done above are each defined by the economic entity creating a report. So, for example, the balance sheet structure for

---

[169] Accounting Equation, Very Basic, http://xbrlsite.azurewebsites.net/2020/introduction/ae-basic/ae_ModelStructure.html

[170] Accounting Equation with Net Assets representation, http://xbrlsite.azurewebsites.net/2020/introduction/ae-na/ae_ModelStructure.html

Microsoft might be something like "**http://microsoft.com/role/BalanceSheet**" and Apple might define their balance sheet using something like "**http://apple.com/role/BalanceSheet**" and you would be able to identify the balance sheet of Microsoft and the balance sheet of Apple; but there is no way to identify the notion of a balance sheet to a machine. Just keep this notion in the back of your mind for now and the ramifications of what we are describing here will become more apparent in a few moments. To a human, those identifiers might be distinguishable as describing the same thing. But to a computer, those are two different names and some mechanism must be used to help a computer understand that they are in fact describing the same idea "balance sheet".

The simple idea of the accounting equation is helpful here to communicate some basic ideas, but we will now move on with a slightly larger financial report structure, the SFAC 6 Elements of Financial Statements, so we can understand a few more moving pieces of an XBRL-based financial report. Again, while SFAC 6 is a simple system it is perfectly capable of getting the important ideas across.

6.4. SFAC 6 Elements of Financial Statements

As we pointed out in the introduction SFAC 6 Elements of Financial Statements includes three structures: balance sheet, income statement, and changes in equity[171]:

Label	Object Class	Period Type	Balance	Report ElementName
01-Balance Sheet	Network			http://www.xbrlsite.com/sfac6/role/BalanceSheet
Balance Sheet [Arithmetic]	Abstract			sfac6:BalanceSheetArithmetic
Assets	Concept (Monetary)	As Of	Debit	sfac6:Assets
Liabilities	Concept (Monetary)	As Of	Credit	sfac6:Liabilities
Equity	Concept (Monetary)	As Of	Credit	sfac6:Equity
02-Comprehensive Income	Network			http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome
Comprehensive Income [Roll Up]	Abstract			sfac6:ComprehensiveIncomeRollUp
Revenues	Concept (Monetary)	For Period	Credit	sfac6:Revenues
(Expenses)	Concept (Monetary)	For Period	Debit	sfac6:Expenses
Gains	Concept (Monetary)	For Period	Credit	sfac6:Gains
(Losses)	Concept (Monetary)	For Period	Debit	sfac6:Losses
Comprehensive Income	Concept (Monetary)	For Period	Credit	sfac6:ComprehensiveIncome
04-Changes in Equity	Network			http://www.xbrlsite.com/sfac6/role/ChangesInEquity
Changes in Equity [Roll Forward]	Abstract			sfac6:ChangesInEquityRollForward
Equity, Beginning Balance	Concept (Monetary)	As Of	Credit	sfac6:Equity
Comprehensive Income	Concept (Monetary)	For Period	Credit	sfac6:ComprehensiveIncome
Investments by Owners	Concept (Monetary)	For Period	Credit	sfac6:InvestmentsByOwners
(Distributions to Owners)	Concept (Monetary)	For Period	Debit	sfac6:DistributionsToOwners
Equity, Ending Balance	Concept (Monetary)	As Of	Credit	sfac6:Equity

So, we have three distinct structures as can be observed above.

However, a close reading of the definitions of the terms that make up SFAC 6, you will recognize that a second approach to representing an income statement is permissible which separates income into two groups: income from normal activities and income from peripheral or incidental activities. This is represented by adding a second income statement structure as can be seen below[172]:

---

[171] SFAC 6 Elements of Financial Statements, http://xbrlsite.azurewebsites.net/2020/master/sfac6/index.html

[172] SFAC 6 which adds second income statement, http://xbrlsite.azurewebsites.net/2020/introduction/sfac6-dim/sfac6_ModelStructure.html

Label	Object Class	Period Type	Balance	Report ElementName
01-Balance Sheet	Network			http://www.xbrlsite.com/sfac6/role/BalanceSheet
Balance Sheet [Arithmetic]	Abstract			sfac6:BalanceSheetArithmetic
Assets	Concept (Monetary)	As Of	Debit	sfac6:Assets
Liabilities	Concept (Monetary)	As Of	Credit	sfac6:Liabilities
Equity	Concept (Monetary)	As Of	Credit	sfac6:Equity
02-Comprehensive Income	Network			http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome
Comprehensive Income [Roll Up]	Abstract			sfac6:ComprehensiveIncomeRollUp
Revenues	Concept (Monetary)	For Period	Credit	sfac6:Revenues
(Expenses)	Concept (Monetary)	For Period	Debit	sfac6:Expenses
Gains	Concept (Monetary)	For Period	Credit	sfac6:Gains
(Losses)	Concept (Monetary)	For Period	Debit	sfac6:Losses
Comprehensive Income	Concept (Monetary)	For Period	Credit	sfac6:ComprehensiveIncome
03-Comprehensive Income 2	Network			http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome2
Comprehensive Income [Roll Up]	Abstract			sfac6:ComprehensiveIncomeRollUp
Income from Normal Activities of Entity	Concept (Monetary)	For Period	Credit	sfac6:IncomeFromNormalActivitiesOfEntity
Income from Peripheral or Incidental Transactions of Entity	Concept (Monetary)	For Period	Credit	sfac6:IncomeFromPeripheralOrIncidentalTransactionsOfEntity
Comprehensive Income	Concept (Monetary)	For Period	Credit	sfac6:ComprehensiveIncome
04-Changes in Equity	Network			http://www.xbrlsite.com/sfac6/role/ChangesInEquity
Changes in Equity [Roll Forward]	Abstract			sfac6:ChangesInEquityRollForward
Equity, Beginning Balance	Concept (Monetary)	As Of	Credit	sfac6:Equity
Comprehensive Income	Concept (Monetary)	For Period	Credit	sfac6:ComprehensiveIncome
Investments by Owners	Concept (Monetary)	For Period	Credit	sfac6:InvestmentsByOwners
(Distributions to Owners)	Concept (Monetary)	For Period	Debit	sfac6:DistributionsToOwners
Equity, Ending Balance	Concept (Monetary)	As Of	Credit	sfac6:Equity

As professional accountants understand, financial reports are not static forms. The conceptual frameworks of both US GAAP and IFRS have the notion of intermediate components which can be different totals and subtotals. In these two income statement examples, the total "Comprehensive Income" is summarized in two different ways. Each of these ways is fundamentally a different income statement format.

And so, above you see two income statements. A careful reading of SFAC 6[173] makes it clear that each of the concepts Revenues, Expenses, Gains, and Losses could include either from normal activities or peripheral/incidental activities so the two statements are correctly disjointed.

Further, SFAC 6 discusses net assets and therefore it would be very appropriate to add another structure for the notion of a statement of net assets. We will not make that addition here but will do so later to try and keep the examples as small as possible.

We talked about the notion of an XBRL network and how they are used to separate structures because you want to or because you have to in order to avoid conflicts. XBRL has another mechanism for creating structures which is called the hypercube which we will introduce now.

Note that the IFRS Conceptual Framework is similar in their explanation of these dynamics, they simply use slightly different terms[174]. In fact, these dynamics are the same for every financial reporting framework[175].

6.5. More Information about Defining Structures using Hypercubes

We introduced you to the notion of a hypercube and how hypercubes can be used to define structures or must be used to define structures when you need additional noncore dimensions. The variance analysis, segment breakdown, and prior period adjustment are all cases where you must use hypercubes.

---

[173] FASB, SFAC 6 Elements of Financial Statements, https://www.fasb.org/pdf/con6.pdf

[174] IFRS.ORG, Conceptual Framework of Financial Reporting, https://www.ifrs.org/issued-standards/list-of-standards/conceptual-framework/

[175] Comparison of Elements of Financial Statements, http://xbrlsite.azurewebsites.net/2020/master/ElementsOfFinancialStatements.pdf

To round out the three different approaches to defining a structure there is also the approach of using a combination of both a **Network and a Hypercube** to define a structure.

The first question you might have is, "Why three approaches?" That is a very good question but providing a good answer is harder. For now, don't worry about the answer to the question because creating XBRL-based reports using IFRS or US GAAP allows either of the three approaches. The "best approach" can be a matter of opinion and circumstances.

Both a Hypercube and a Network are ways of connecting associations together to define a structure. Networks must always exist in XBRL-based reports. Hypercubes are optional until you need what only they can provide to structure information effectively.

Essentially, Networks and Hypercubes connect sets of associations and facts together and distinguish one set of associations with another set. What we try and do is make the notion of a Network and Hypercube invisible to you and expose only the notion of the Structure itself. This is done by automatically generating the Network, Hypercube, or combination of Network + Hypercube by cleverly designing software. If defining hypercubes are automatic, then using hypercubes becomes somewhat invisible to an accountant creating an XBRL-based report and that is one less decision that needs to be made or thing that needs to be understood.

But, let's forget about all that for now and look at a Hypercube. We do this by looking at the following SFAC 6 representation which for simplicity reverts back to a single income statement: [176]

Report Element Label	Report Element Category	Period Type	Balance Type	Report Element Name
01-Balance Sheet	Network			http://www.xbrlsite.com/sfac6/role/BalanceSheet
Balance Sheet [Hypercube]	Hypercube			sfac6:BalanceSheetHypercube
Balance Sheet [Line Items]	LineItems			sfac6:BalanceSheetLineItems
Balance Sheet [Arithmetic]	Abstract			sfac6:BalanceSheetArithmetic
Assets	Concept (Monetary)	As Of	Debit	sfac6:Assets
Liabilities	Concept (Monetary)	As Of	Credit	sfac6:Liabilities
Equity	Concept (Monetary)	As Of	Credit	sfac6:Equity
02-Comprehensive Income	Network			http://www.xbrlsite.com/sfac6/role/ComprehensiveIncome
Comprehensive Income Statement [Hypercube]	Hypercube			sfac6:ComprehensiveIncomeStatementHypercube
Comprehensive Income Statement [Line Items]	LineItems			sfac6:ComprehensiveIncomeStatementLineItems
Comprehensive Income [Roll Up]	Abstract			sfac6:ComprehensiveIncomeRollUp
Revenues	Concept (Monetary)	For Period	Credit	sfac6:Revenues
(Expenses)	Concept (Monetary)	For Period	Debit	sfac6:Expenses
Gains	Concept (Monetary)	For Period	Credit	sfac6:Gains
(Losses)	Concept (Monetary)	For Period	Debit	sfac6:Losses
Comprehensive Income	Concept (Monetary)	For Period	Credit	sfac6:ComprehensiveIncome
03-Changes in Equity	Network			http://www.xbrlsite.com/sfac6/role/ChangesInEquity
Changes in Equity [Hypercube]	Hypercube			sfac6:ChangesInEquityHypercube
Changes in Equity [Line Items]	LineItems			sfac6:ChangesInEquityLineItems
Changes in Equity [Roll Forward]	Abstract			sfac6:ChangesInEquityRollForward
Equity, Beginning Balance	Concept (Monetary)	As Of	Credit	sfac6:Equity
Comprehensive Income	Concept (Monetary)	For Period	Credit	sfac6:ComprehensiveIncome
Investments by Owners	Concept (Monetary)	For Period	Credit	sfac6:InvestmentsByOwners
(Distributions to Owners)	Concept (Monetary)	For Period	Debit	sfac6:DistributionsToOwners
Equity, Ending Balance	Concept (Monetary)	As Of	Credit	sfac6:Equity

A hypercube is a type of XBRL **report element**. Up until now we have not really discussed the different types of XBRL report elements because there was only one type, the concept, which could or could not be abstract. Along with the *hypercube* comes two additional types of report elements: the *dimension* and the *member*.

A hypercube is a structure which combines a set of dimensions, members, and concepts to represent some set of information within an XBRL-based financial report.

---

[176] SFAC 6 Adding Hypercubes, http://xbrlsite.azurewebsites.net/2020/intermediate/sfac6/sfac6_ModelStructure.html

For now, we will briefly introduce the notion of the XBRL report element the fact that there are seven different categories of such report elements, and that the pattern of how those report elements can be arranged within a structure is explained by the following matrix:

		Parent						
		Network	Table	Axis	Member	Line Items	Abstract	Concept
Child	Network	Illegal XBRL	Illegal XBRL	Illegal XBRL	Illegal XBRL	Illegal XBRL	Illegal XBRL	Illegal XBRL
	Table	OK	Disallowed	Disallowed	Disallowed	Disallowed	OK	Disallowed
	Axis	Disallowed	OK	Disallowed	Disallowed	Disallowed	Disallowed	Disallowed
	Member	Disallowed	Disallowed	OK	OK	Disallowed	Disallowed	Disallowed
	Line Items	Disallowed	OK	Disallowed	Disallowed	Disallowed	Disallowed	Disallowed
	Abstract	OK	Disallowed	Disallowed	Disallowed	OK	OK	Disallowed
	Concept	Disallowed	Disallowed	Disallowed	Disallowed	OK	OK	Disallowed

A report element can be a parent and/or it could be a child in an association. The matrix simply summarizes the appropriate and inappropriate parent/child associations of the different categories of report elements. Note that a "Table" is an alias for the term Hypercube and that "Axis" is an alias for the term Dimension. The US GAAP and IFRS XBRL taxonomies use the terms Table and Axis as opposed the terms Hypercube and Dimension to describe those two categories of report elements. We are going to use the terms Hypercube and Dimension throughout this document.

Let us leave report elements for now, we will explain them in more detail as they are introduced in our modeling of different financial reporting related models.

So, to summarize what you see in the representation of SFAC 6 above; you have both a uniquely named network and a uniquely named hypercube that are used to represent each structure that we use to represent SFAC 6: balance sheet, income statement, changes in equity.

Is this the only approach? No. Let's now look at an alternative approach to representing exactly the same set of information from SFAC 6 and let's also add back that second income statement structure.

6.6. Mathematical Connections Between Proof Information Structures

The Common Elements of Financial Statements representation was specifically referenced to make the point that is communicated in both the SFAC 6 conceptual framework and IFRS conceptual framework which is the notation of articulation. Articulation is the notion that the primary financial statements are connected together mathematically[177]:

---

[177] Articulation of Four Statement Model, http://xbrlsite.azurewebsites.net/2020/core/master-common/ArticulationFourStatementModel.jpg

Expanding this notion of articulation using the additional mathematical associations of the PROOF representation it becomes apparent that the information fragments represented are correctly represented because the model is completely intact mathematically as is shown below[178]:

Specifically note that:

1. The prior period adjustment properly connects to the beginning balance of the changes in equity roll forward.

2. The segment revenues member aggregation properly connects to the income statement roll up.

3. The variance analysis actual amounts tie to the income statement roll up and the variance between actual and budget foot and cross cast.

This proves that the XBRL-based representation of financial report information is appropriate because that representation is confirmed by the global standard XBRL and rules processors show they are consistent with expectation.

6.7. Understanding Intersections between Structures and "Dimension Defaults"

As pointed out in the prior section one structure can be intersected with one or more other structures. For example, there is a relation between the structure which breaks down segment Revenues by segment and the structure that represents the subcomponents of Net Income which also contains the fact Revenues:

---

[178] Articulation of Proof representation mathematical relations,
http://xbrlsite.azurewebsites.net/2021/essentials/Proof/ProofMathematicalComputations.jpg

The fact Revenues for the period 2020-01-01 to 2020-12-31 exists only once in the report, but it is used to represent information in both structures. Notice that the dimensions of the fact Revenues is different in the two structures. Whereas in the income statement income breakdown there is no "Segments [Axis]" but in the segment revenues breakdown that dimension does exist. See that here by comparing these two fact tables.

Income statement[179]:

Component: (Network and Table)	
Network	**03-Income Statement** (http://www.xbrlsite.com/proof/role/ComprehensiveIncome)
Table	Comprehensive Income Statement [Hypercube]

#	Reporting Entity [Axis]	Period [Axis]	Concept	Fact Value	Unit	Rounding
1	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Revenues	7000	USD	INF
2	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Expenses	3000	USD	INF
3	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Gains	1000	USD	INF
4	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Losses	2000	USD	INF
5	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Net Income	3000	USD	INF

Segment revenues[180]:

Component: (Network and Table)	
Network	**12-Segment Revenues** (http://www.xbrlsite.com/proof/role/SegmentRevenues)
Table	Segment Revenues [Hypercube]

#	Reporting Entity [Axis]	Period [Axis]	Segments [Axis]	Concept	Fact Value	Unit	Rounding
1	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	All Segments [Member]	Revenues	7000	USD	INF
2	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Segment Bravo [Member]	Revenues	4000	USD	INF
3	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Segment Alpha [Member]	Revenues	1000	USD	INF
4	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Segment Charlie [Member]	Revenues	2000	USD	INF

Here is what is going on?

---

[179] Income statement fact table, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/evidence-package/contents/index.html#NetworkFacts-ComprehensiveIncome-proof_ComprehensiveIncomeStatementHypercube.html

[180] Segment revenues fact table, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/evidence-package/contents/index.html#NetworkFacts-SegmentRevenues-proof_SegmentRevenuesHypercube.html

If you examine the XBRL instance, the fact distinguished as representing Revenues for the period distinguished as being 2020-01-01 to 2020-12-31 exists in that XBRL instance only once. That fact "morphs" into the form it needs to be in, with or without that Segments [Axis] dimension using a mechanism named the "dimension-default". Many people misinterpret the function of the dimension-default as there to save you time from having to enter a dimension. But that is not what is going on. What is going on is the dimension-default mechanism tells the fact how to morph in order to work correctly in both structures.

Proper morphing the dimensions to suit both hypercubes in the example structures above shows you how dimension-defaults are used to virtually indicate information that is not physically represented within the XBRL instance context information for an explicit dimension.

Further note that the Revenues fact as used in the variance analysis fact table has yet another different dimension, Scenario [Axis][181]:

#	Reporting Entity [Axis]	Period [Axis]	Scenario [Axis]	Concept	Fact Value	Unit	Rounding
1	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Actual [Member]	Revenues	7000	USD	INF
2	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)	2020-01-01 - 2020-12-31	Variance [Member]	Revenues	1000	USD	INF

6.8. Difference between Signal and Noise

Constructing hypercubes effectively is a learned skill. There is a big difference between providing "signal" which is more information and "noise" which is unimportant stuff that just gets in the way[182]. Contrast the use of dimensions on these hypercubes:

Here is the financial highlights of the Proof Baseline to use as a starting point for our comparison[183]:

Component: (Network and Table)	
Network	**14-Financial Highlights** (http://www.xbrlsite.com/proof/role/FinancialHighlights)
Table	Financial Highlights [Hypercube]

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

Reporting Entity [Axis]	GH259400TOMPUOLS65II (http://standards.iso.org/iso/17442)

	Period [Axis]
Financial Highlights [Line Items]	2020-01-01 - 2020-12-31
Financial Highlights [Set]	
Revenues	7,000
Comprehensive Income	3,000
Distributions to Owners	500

This representation is very simple. There are no additional noncore dimensions provided, but there is a hypercube being used. Note that you do not need to provide dimensions when you use a hypercube.

---

[181] Variance analysis fact table, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/evidence-package/contents/index.html#NetworkFacts-VarianceAnalysis-proof_VarianceAnalysisHypercube.html

[182] Representing Information Logically, http://xbrl.squarespace.com/journal/2020/12/30/representing-information-logically.html

[183] Proof, Financial Highlights, http://xbrlsite.azurewebsites.net/2021/essentials/Proof/evidence-package/contents/index.html#Rendering-FinancialHighlights-proof_FinancialHighlightsHypercube.html

Here is a slightly more realistic set of financial highlights that contains a few additional facts and five periods rather than just one; but the structure still does not make use of dimensions[184]:

Component: (Network and Table)					
Network	**AA: Statement: Financial Highlights** (http://www.ABCCompany.com/company/role/level4/FinancialHighlightsSchedule)				
Table	Financial Highlights [Table]				

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

Reporting Entity [Axis]	10810137d58f76b84aaa (http://standards.iso.org/iso/17442)

	Period [Axis]				
Financial Highlights [Line Items]	2020-01-01 - 2020-12-31	2019-01-01 - 2019-12-31	2018-01-01 - 2018-12-31	2017-01-01 - 2017-12-31	2016-01-01 - 2016-12-31
Financial Highlights [Hierarchy]					
Revenues, Net	4,000,000	5,000,000	4,000,000	4,000,000	4,000,000
Income (Loss) from Continuing Operations	500,000	(4,000,000)	(4,000,000)	(4,000,000)	(4,000,000)
Net Income (Loss)	500,000	(4,000,000)	(4,000,000)	(4,000,000)	(4,000,000)
Cash Flow Provided by (used in) Operating Activities, Net	(1,000,000)	4,000,000	0	0	0
Capital Additions	750,000	650,000	550,000	450,000	350,000
Average Number of Employees	300	250	250	240	220

It is relatively easy to transition between the first and second representation of financial highlights.

Here is another representation of financial highlights, this time using three noncore dimensions; Legal Entity [Axis], Report Date [Axis], and Reporting Scenario [Axis]; each of which uses the dimension default member as the value of the dimension[185]:

Component: (Network and Table)					
Network	**AA: Statement: Financial Highlights** (http://www.ABCCompany.com/company/role/level4/FinancialHighlightsSchedule)				
Table	Financial Highlights [Table]				

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

Reporting Entity [Axis]	10810137d58f76b84aaa (http://standards.iso.org/iso/17442)
Legal Entity [Axis]	Consolidated Entity [Member]
Report Date [Axis]	Reported as of March 18, 2021 [Member]
Reporting Scenario [Axis]	Actual [Member]

	Period [Axis]				
Financial Highlights [Line Items]	2020-01-01 - 2020-12-31	2019-01-01 - 2019-12-31	2018-01-01 - 2018-12-31	2017-01-01 - 2017-12-31	2016-01-01 - 2016-12-31
Financial Highlights [Hierarchy]					
Revenues, Net	4,000,000	5,000,000	4,000,000	4,000,000	4,000,000
Income (Loss) from Continuing Operations	500,000	(4,000,000)	(4,000,000)	(4,000,000)	(4,000,000)
Net Income (Loss)	500,000	(4,000,000)	(4,000,000)	(4,000,000)	(4,000,000)
Cash Flow Provided by (used in) Operating Activities, Net	(1,000,000)	4,000,000	0	0	0
Capital Additions		650,000	550,000	450,000	350,000
Average Number of Employees	300	250	250	240	220

What is the difference between the logic provided in the first and second representations of the XASB representation? The answer is that logically, both of these convey exactly the same meaning although the second explicitly provides those three noncore dimensions information and the first does not.

---

184 XASB representation, financial highlights, no dimensions,
http://xbrlsite.azurewebsites.net/2021/essentials/Dimensions/dimensions-none/evidence-package/contents/index.html#Rendering-FinancialHighlightsSchedule-gaap_FinancialHighlightsTable.html

185 XASB representation, financial highlights, three noncore dimensions added,
http://xbrlsite.azurewebsites.net/2021/essentials/Dimensions/dimensions-moderate/evidence-package/contents/index.html#Rendering-FinancialHighlightsSchedule-gaap_FinancialHighlightsTable.html

Let us take this a bit farther. Here you see the same information conveyed in the previous two XASB representations but here several more dimensions are added so that we now have 11 noncore dimensions[186]:

Component: (Network and Table)	
Network	**AA: Statement: Financial Highlights** (http://www.ABCCompany.com/company/role/level4/FinancialHighlightsSchedule)
Table	Financial Highlights [Table]

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

Reporting Entity [Axis]	10810137d58f76b84aaa (http://standards.iso.org/iso/17442)
Legal Entity [Axis]	Consolidated Entity [Member]
Report Date [Axis]	Reported as of March 18, 2021 [Member]
Reporting Scenario [Axis]	Actual [Member]
Property, Plant and Equipment, Description of Major Class [Axis]	Property, Plant and Equipment, All Major Classes [Member]
Customer [Axis]	All Customers [Member]
Debt Instrument, Identifier [Axis]	All Debt Instruments [Member]
Business Segment [Axis]	Business Segments, All [Member]
Related Party, Name [Axis]	All Related Parties [Member]
Director Name [Axis]	All Directors [Member]
Share Ownership Plan, Identifier [Axis]	All Share Ownership Plans [Member]
Reconciling Item, Type [Axis]	All Reconciling Items of Cash and Cash Equivalents [Member]

	Period [Axis]				
Financial Highlights [Line Items]	2020-01-01 - 2020-12-31	2019-01-01 - 2019-12-31	2018-01-01 - 2018-12-31	2017-01-01 - 2017-12-31	2016-01-01 - 2016-12-31
Financial Highlights [Hierarchy]					
Revenues, Net	4,000,000	5,000,000	4,000,000	4,000,000	4,000,000
Income (Loss) from Continuing Operations	500,000	(4,000,000)	(4,000,000)	(4,000,000)	(4,000,000)
Net Income (Loss)	500,000	(4,000,000)	(4,000,000)	(4,000,000)	(4,000,000)
Cash Flow Provided by (used in) Operating Activities, Net	(1,000,000)	4,000,000	0	0	0
Capital Additions		650,000	550,000	450,000	350,000
Average Number of Employees	300	250	250	240	220

We already established that the last two representations of financial highlights were logically the same; so what about now. What does adding the additional eight dimensions get you?

Logically, the three XASB representations are identical. Running each through an XBRL Formula processor shows this and that each of the representations is valid XBRL technical syntax.

Yes, adding dimensions can make more information explicitly available than necessary to derive. However, this comes at a cost. The more dimensions you add that are not relevant the harder it can be to interpret the reported information.

Balancing the needs of humans and the needs of computers reading information is a learned skill.

7. Advanced (Other Important Odds and Ends)

I don't have a good title for this section yet, I will figure that out over time. Here are some additional important odds and ends that need to be understood and consciously considered.

---

[186] XASB representation, financial highlights, eleven noncore dimensions, http://xbrlsite.azurewebsites.net/2021/essentials/Dimensions/dimensions-many/evidence-package/contents/index.html#Rendering-FinancialHighlightsSchedule-gaap_FinancialHighlightsTable.html

7.1. Patterns and Models are Logical Rules

Why all the work to identify and understand patterns and models? Well, because patterns and models provide logical rules; in fact it provides a logical schema[187]. These patterns and rules provide a specification for what is permissible per XBRL-based financial reporting.

These patterns and models are documented in the *Logical Theory Describing Financial Report*[188]. Software engineers have a tendency to implement the XBRL technical specification verbatim. While that will work, it makes implemented software applications harder to use because you need to learn about the XBRL technical specification. Leveraging higher level models makes software both easier to implement and easier to use.

Software engineers use the rules provided by patterns and models to build software that is easy and straight forward to use.

7.2. Software Interoperability

Software interoperability, ease of use, completeness of functionality, and capability to prove that an XBRL-based financial report is a properly functioning logical system is important to effectively exchanging information using the XBRL technical format. Here are four software applications that support this method:

- Pesseract: http://xbrlsite.azurewebsites.net/2020/master/Pesseract.html
- Pacioli: http://xbrlsite.azurewebsites.net/2020/master/Pacioli.html
- XBRL Query: http://xbrlsite.azurewebsites.net/2020/master/XBRLQuery.html
- XBRL Cloud: http://xbrlsite.azurewebsites.net/2020/master/XBRLCloud.html
- Auditchain Suite: https://dev.auditchain.finance/

There is likely other software that can be used for XBRL-based digital financial reporting, I am simply pointing out software that I have rigorously tested.

7.3. Identifying Disclosures

When we work with a financial report, we generally work with the fragments that make up the report. As I have pointed out we can break a full report into the individual fragments that make up that report. Those fragments can be broken and identified by humans as to the individual disclosures that comprise a particular financial report. For example, a financial report can be broken down into the balance sheet, income statement, cash flow statement, statement of changes in equity, individual policies, and individual detailed disclosures that are included within the notes of a financial statement.

It is of critical important to differentiate the notion of a "note" and a "disclosure" in your mind. A disclosure is an informational artifact and indicates information that is being disclosed. A note is a presentational artifact and is comprised of a set of one to many disclosures. Accountants can somewhat arbitrarily organize disclosures into notes. But the disclosures themselves are generally prescribed and are driven by the

---

[187] Need for a Logical Schema, https://digitalfinancialreporting.blogspot.com/2023/06/need-for-logical-schema.html
[188] Logical Theory Describing Financial Report, http://xbrl.squarespace.com/logical-theory-financial-rep/

information provided by the disclosure rather than what note a disclosure ends up being located within. Also, accountants may have different alternatives as to what form a disclosure can take and even whether a disclosure is required/necessary or not in some cases. But what makes up an individual disclosure is generally prescribed.

Accountants can work with these disclosures and even communicate with one another about individual disclosures fairly effectively. But computers cannot work with disclosures effectively unless you give each disclosure a unique name that identifies each individual disclosure.

We have done this for the disclosures relating to the PROOF representation[189]. You can get to those disclosures in machine readable form here[190]:

proof

Entry Point (Associations) | Terms | Structures | Rules | Type-subtype | Disclosures | Dislosure Mechanics Rules | Dislosure Rules (Reporting Checklist)

#	Report Element Label	Report Cate
1	01-Balance Sheet	Network
2	**Balance Sheet [Hypercube]**	Hypercube
3	**Balance Sheet [Line Items]**	LineItems
4	Assets [Roll Up]	Abstract
5	Current Assets	Concept (Mone
6	Noncurrent Assets	Concept (Mone
7	Assets	Concept (Mone

If you open that machine-readable list of disclosures for the PROOF representation which is represented using the XBRL technical syntax and shown below using standard XBRL software, you can read that list:

Prefix	Label	Name	Data Type
disclosures	Assets Roll Forward	AssetsRollForward	disclosureItemType
disclosures	Assets Roll Up	AssetsRollUp	disclosureItemType
disclosures	Balance Sheet	BalanceSheet	disclosureItemType
disclosures	Basis of Reporting	BasisOfReporting	disclosureItemType
disclosures	Cash Flow Statement	CashFlowStatement	disclosureItemType
disclosures	Changes in Equity	ChangesInEquity	disclosureItemType
disclosures	Financial Highlights	FinancialHighlights	disclosureItemType
disclosures	Income Statement	IncomeStatement	disclosureItemType
disclosures	Income Statement Alternative	IncomeStatementAlterna...	disclosureItemType
disclosures	Liabilities and Equity Roll Up	LiabilitiesAndEquityRollUp	disclosureItemType
disclosures	Nature of Operations	NatureOfOperations	disclosureItemType
disclosures	Net Assets Roll Up	NetAssetsRollUp	disclosureItemType
disclosures	Net Cash Flow Roll Up	NetCashFlowRollUp	disclosureItemType
disclosures	Prior Period Errors	PriorPeriodErrors	disclosureItemType
disclosures	Revenue Recognition Policy	RevenueRecognitionPolicy	disclosureItemType
disclosures	Segment Revenues	SegmentRevenues	disclosureItemType
disclosures	Stock Plan Activity	StockPlanActivity	disclosureItemType
disclosures	Variance Analysis	VarianceAnalysis	disclosureItemType

What is more important is that a computer can read that list also.

---

[189] Proof representation, http://xbrlsite.azurewebsites.net/2020/intermediate/proof/proof_ModelStructure.html
[190] Proof disclosures, http://xbrlsite.azurewebsites.net/2020/intermediate/proof/disclosures.xsd

Think about this for a moment. Have you ever seen a complete list of disclosures before? Probably not. No one has really bothered to create such a list. But what if where were such a list of disclosure names. What if that same list of names was uses by intermediate accounting textbooks, the authoritative literature, interpretations of the authoritative literature, disclosure checklists used to create financial reports, the preparers of financial reports, analysts extracting information from financial reports, etc. You probably get the point. None of the value of a complete, reliable, and machine-readable list is available until someone creates the list and puts that information into machine readable form. Here[191] is that list for US GAAP which contains 995 disclosures which have been named (a working prototype). Here[192] is a human readable version of that same list that is searchable and filterable.

And so, each disclosure for US GAAP can be named and 995 such named disclosures are provided for US GAAP in the working prototype above. Similar lists can be created for IFRS[193] or any financial reporting scheme really.

For example, one disclosure name for US GAAP is "**InventoryNetRollUp**". The specific details of that disclosure are also described[194] including as an example a specific permissible prototype of the disclosure:

Line		Object Class (Data type)	Period Type	Balance	Report Element Name
1	*Inventory balances*	Abstract			us-gaap:InventoryRawMaterialsAndSuppliesNetOfReservesAbstract
2	Raw materials and packaging	Concept (Monetary)	As Of	Debit	us-gaap:InventoryRawMaterialsAndSuppliesNetOfReserves
3	Finished goods	Concept (Monetary)	As Of	Debit	us-gaap:InventoryFinishedGoodsNetOfReserves
4	Other	Concept (Monetary)	As Of	Debit	us-gaap:OtherInventoryNetOfReserves
5	Total inventories	Concept (Monetary)	As Of	Debit	us-gaap:InventoryNet

What is more, examples of actual disclosures provided by economic entities are available and identifiable from reports of public companies submitted to the Securities

---

[191] US GAAP Disclosures, machine-readable http://xbrlsite.azurewebsites.net/2020/reporting-scheme/us-gaap/disclosures-topics/disclosures.xsd
[192] US GAAP Disclosures, human-readable, http://xbrlsite.azurewebsites.net/2020/reporting-scheme/us-gaap/documentation/Disclosures.html
[193] IFRS Disclosures, human-readable, http://xbrlsite.azurewebsites.net/2020/reporting-scheme/ifrs/documentation/Disclosures.html
[194] Inventory Net Roll Up disclosure details, http://xbrlsite.azurewebsites.net/2020/reporting-scheme/us-gaap/disclosures-topics/disclosures-detail/Disclosure-517.html

and Exchange Commission (SEC). These are all freely available in both human-readable form and machine-readable form. For example, here is the inventory roll up disclosure (InventoryNetRollUp) provided by Boeing[195]:

Component: (Network and Table)	
Network	2408403 - Disclosure - Inventories (Inventory Disclosure Table) (Details) (http://www.boeing.com/role/InventoriesInventoryDisclosureTableDetails)
Table	(Implied)

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

Reporting Entity [Axis]	0000012927 (http://www.sec.gov/CIK)

Inventory Disclosure [Abstract]	Period [Axis]	
	2013-12-31	2012-12-31
Inventory Disclosure [Abstract]		
Long-term contracts in progress	12,608,000,000	15,130,000,000
Commercial aircraft programs	48,065,000,000	40,389,000,000
Commercial spare parts, used aircraft, general stock materials and other	7,793,000,000	7,206,000,000
Inventory before advances and progress billings	68,466,000,000	62,725,000,000
Less advances and progress billings	(25,554,000,000)	(24,974,000,000)
Total	42,912,000,000	37,751,000,000

In fact, an entire system for researching disclosures can be created by taking the simple step of creating a name and then using that name to specify what disclosure you want to see examples for[196]:

---

[195] Boeing Inventory Disclosure, http://www.xbrlsite.com/2015/fro/us-gaap/html/Disclosures/Exemplars/20982-Exemplar.html

[196] Working Prototype of disclosure research system, http://xbrlsite-app.azurewebsites.net/DisclosureBestPractices_USGAAP/DisclosureBestPractices.aspx?DisclosureName=InventoryNetRollUp

All this shows you the power of metadata and what can be achieved by the simple step of giving each disclosure a unique name so that you can refer to any disclosure in a manner that a software application can understand what you are talking about and do this for any financial reporting scheme!

But how do you use a name to discover a disclosure? Well, you don't use the name. You use other information that you attach to the name to help you discover the report fragment that discloses specific information.

7.4. Disclosure Mechanics

Disclosure mechanics is a set of machine-readable rules represented using XBRL[197] that describes how a specific disclosure is represented within an XBRL-based financial report. You can use that information for two things. First, you can use the information to detect the disclosure within the report. Second, you can use the information to verify that the disclosure is represented consistent with your expectations. Here is an example of the disclosure mechanics validation results of the Proof representation we are using.

---

[197] Disclosure Mechanics Rules for Proof, http://xbrlsite.azurewebsites.net/2020/intermediate/proof/dm.xsd

Primary Information

#	Disclosure	Category	Level	Pattern	Disclosure Found	Disclosure Consistent	Applicable	Representation Concept [TEXT BLOCK]	Representation Concept DETAIL
1	Assets Roll Forward	Unknown	Level4Detail	RollForward	True	CONSISTENT	True	NOT-EXPECTED	Assets
2	Assets Roll Up	Unknown	Level4Detail	RollUp	True	CONSISTENT	True	NOT-EXPECTED	Assets
3	Balance Sheet	Unknown	UNKNOWN	Component	True	CONSISTENT	True	-	-
4	Basis of Reporting	Unknown	Level1TextBlock	TextBlock	True	CONSISTENT	True	Basis of Reporting [Text Block]	NOT-EXPECTED
5	Cash Flow Statement	Unknown	UNKNOWN	Component	True	CONSISTENT	True	-	-
6	Changes in Equity	Unknown	Level4Detail	RollForward	True	CONSISTENT	True	NOT-EXPECTED	Equity
7	Financial Highlights	Unknown	Level4Detail	Hierarchy	True	CONSISTENT	True	NOT-EXPECTED	Revenues
8	Income Statement	Unknown	Level4Detail	RollUp	True	CONSISTENT	True	NOT-EXPECTED	Comprehensive Income
9	Income Statement Alternative	Unknown	Level4Detail	RollUp	True	CONSISTENT	True	NOT-EXPECTED	Comprehensive Income
10	Liabilities and Equity Roll Up	Unknown	Level4Detail	RollUp	True	CONSISTENT	True	NOT-EXPECTED	Liabilities and Equity
11	Nature of Operations	Unknown	Level1TextBlock	TextBlock	True	CONSISTENT	True	Nature of Operations [Text Block]	NOT-EXPECTED
12	Net Assets Roll Up	Unknown	Level4Detail	RollUp	True	CONSISTENT	True	NOT-EXPECTED	Net Assets
13	Net Cash Flow Roll Up	Unknown	Level4Detail	RollUp	True	CONSISTENT	True	NOT-EXPECTED	Net Cash Flow
14	Prior Period Errors	Unknown	Level4Detail	Adjustment	True	CONSISTENT	True	NOT-EXPECTED	Equity
15	Revenue Recognition Policy	Unknown	Level1TextBlock	TextBlock	True	CONSISTENT	True	Revenue Recognition Policy [Text Block]	NOT-EXPECTED
16	Segment Revenues	Unknown	Level4Detail	Hierarchy	True	CONSISTENT	True	NOT-EXPECTED	Revenues
17	Stock Plan Activity	Unknown	Level4Detail	RollForwardInfo	True	CONSISTENT	True	NOT-EXPECTED	Nonvested Fair Value
18	Variance Analysis	Unknown	Level4Detail	RollUp	True	CONSISTENT	True	NOT-EXPECTED	Comprehensive Income

Consider the first disclosure, "Assets Roll Forward". If you click on the plus sign, a human readable version of the rule is shown which appears as follows:

| 1 | Assets Roll Forward | | Unknown | Level4Detail | RollForward | True | | CONSISTENT | True | NOT-EXPECTED | Assets |

Rules   Line of Reasoning

This disclosure disclosures:AssetsRollForward ...

- MUST be represented using the Hypercube/[Table] named: proof:CashFlowHypercube

- MUST be represented as a **Level 4 Disclosure Detail** with the concept arrangement pattern: cm:RollForward

   - cm:RollForward REQUIRES beginning/ending balance proof:Assets

You see the name of the disclosure which is "AssetsRollForward". You see that the disclosure is required to be represented within the "proof:CashFlowHypercube". You see that the disclosure is a Roll Forward. Finally, you see that the disclosure is required to have the concept "proof:Assets" as the instant concept of that Roll Forward information pattern. Of the fragments contained within the Proof XBRL-based report there is only one fragment that has all of these characteristics above which you can see in the screen shot below:
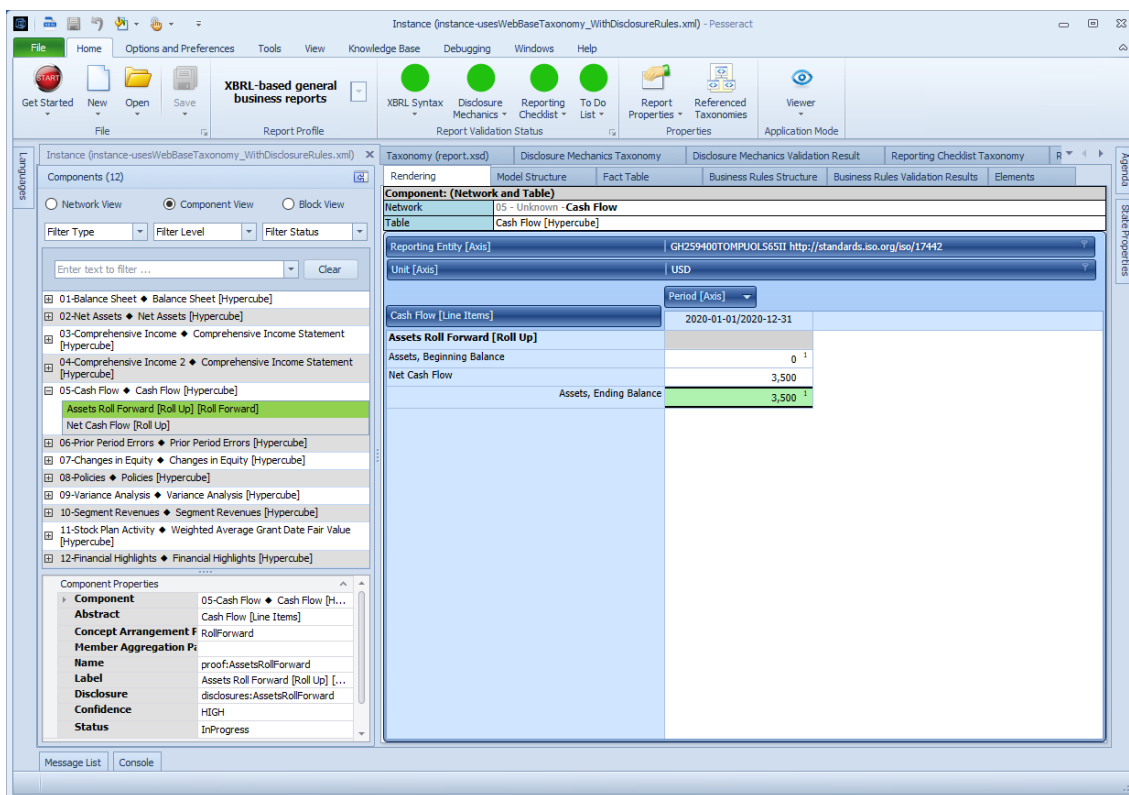
Further, if you press on the "Line of Reasoning" tab you see the line of reasoning used to detect the disclosure:

The rule itself and the line of reasoning can help you have 100% transparency into what a disclosure must look like and why a computer algorithm did or did not locate the disclosure being sought.

In the case of the "Assets Roll Forward" disclosure, all you really needed was the name of the hypercube in order to discover where the disclosure was in the XBRL-based report. The information model of the disclosure (i.e. RollForward) and the required instant concept (i.e. proof:Assets) is not necessary for the disclosure discovery process. However, that information is necessary to determine whether the disclosure has been structured appropriately.

What we are showing for the Proof example might seem simplistic. To give you a better idea of the power of the reasoning software and the rules, consider this example of a US GAAP XBRL Taxonomy rule related to the InventoryNetRollUp disclosure:

There are two important things to notice about the US GAAP XBRL taxonomy rule above. First, notice that no specific hypercube or network is referenced. The disclosure can be effectively discovered without network or hypercube information. Second, notice that other disclosures that should exist if the inventory is found are also queried for such as the inventory policy text block (level 2 text block), the disclosure text block (level 3 disclosure text block), and the note text block (level 1 note text block).

Effectively, the rules engine of software applications can use prototype theory to discover report information even if disclosures are not specifically identifiable via network or hypercube names. Prototype theory is simply a technique used to fined something based on the characteristics of the thing you are looking for instead of a uniquely identifiable name. For more information on prototype theory, see Understanding Prototype Theory[198].

Note that not all information about a disclosure is provided, only the essential information that identifies each disclosure. More and more information can be added to differentiate one disclosure from another.

---

[198] *Understanding Prototype Theory*, http://xbrl.squarespace.com/journal/2013/12/21/understanding-prototype-theory.html

7.5. Reporting Checklist

Professional accountants use what they call a reporting checklist or disclosure checklist as a memory jogger for making sure a financial report is correctly created[199]. A similar strategy can be used to make sure an XBRL-based financial report is correctly created. Here are the machine readable disclosure rules of the proof representation[200] and what those rules look like in human readable form:

-		Arcrole	Order
⌄ D Definition View			
⌄ ◆ Reporting Checklist			
⌄ 🔸 [Thing]			0
⌄ 🔸 [FinancialReport]		http://xbrlsite.azurewebsites.net/2016/conceptual-model/arcrole/class-equivalentClass	1
🔸 Balance Sheet		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-requiresDisclosure	2
🔸 Income Statement		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-requiresDisclosure	3
🔸 Cash Flow Statement		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-requiresDisclosure	3.1
🔸 Changes in Equity		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-requiresDisclosure	4
🔸 Financial Highlights		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	5
🔸 Prior Period Errors		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	6
🔸 Variance Analysis		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	7
🔸 Segment Revenues		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	8
🔸 Stock Plan Activity		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	9
🔸 Basis of Reporting		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-requiresDisclosure	10
🔸 Nature of Operations		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-requiresDisclosure	11
🔸 Revenue Recognition Policy		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-requiresDisclosure	12
🔸 Assets Roll Up		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	21
🔸 Liabilities and Equity Roll Up		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	21
🔸 Net Assets Roll Up		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	22
🔸 Assets Roll Forward		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	23
🔸 Income Statement Alternative		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	24
🔸 Net Cash Flow Roll Up		http://xbrlsite.azurewebsites.net/2016/conceptual-model/drules-arcroles/arcrole/financialReport-possibleDisclosure	25

There are the validation results from running the reporting checklist rules against the Proof representation:

#		Disclosure	Checklist Category	Reason Disclosure Must Exist	Discovered	Expectation Met	Link to Disclosure Mechanics
⌄ 0		Reporting Checklist					
⌄ 1		Balance Sheet	Required disclosure	Disclosure always required, satisfied by Assets Roll Up and Liabilities and Equity Roll Up disclosures	True	CONSISTENT	Balance Sheet
	2	Assets Roll Up	Part of disclosure	Satisfies Balance Sheet disclosure	True	CONSISTENT	Assets Roll Up
	3	Liabilities and Equity Roll Up	Part of disclosure	Satisfies Balance Sheet disclosure	True	CONSISTENT	Liabilities and Equity Roll Up
4		Income Statement	Required disclosure	Disclosure always required	True	CONSISTENT	Income Statement
⌄ 5		Cash Flow Statement	Required disclosure	Disclosure always required, satisfied by Assets Roll Forward and Net Cash Flow Roll Up disclosures	True	CONSISTENT	Cash Flow Statement
	6	Assets Roll Forward	Part of disclosure	Satisfies Cash Flow Statement disclosure	True	CONSISTENT	Assets Roll Forward
	7	Net Cash Flow Roll Up	Part of disclosure	Satisfies Cash Flow Statement disclosure	True	CONSISTENT	Net Cash Flow Roll Up
8		Changes in Equity	Required disclosure	Disclosure always required	True	CONSISTENT	Changes in Equity
9		Financial Highlights	Possible disclosure	Disclosure is present	True	CONSISTENT	Financial Highlights
10		Prior Period Errors	Possible disclosure	Disclosure is present	True	CONSISTENT	Prior Period Errors
11		Variance Analysis	Possible disclosure	Disclosure is present	True	CONSISTENT	Variance Analysis
12		Segment Revenues	Possible disclosure	Disclosure is present	True	CONSISTENT	Segment Revenues
13		Stock Plan Activity	Possible disclosure	Disclosure is present	True	CONSISTENT	Stock Plan Activity
14		Basis of Reporting	Required disclosure	Disclosure always required	True	CONSISTENT	Basis of Reporting
15		Nature of Operations	Required disclosure	Disclosure always required	True	CONSISTENT	Nature of Operations
16		Revenue Recognition Policy	Required disclosure	Disclosure always required	True	CONSISTENT	Revenue Recognition Policy
17		Net Assets Roll Up	Possible disclosure	Disclosure is present	True	CONSISTENT	Net Assets Roll Up
18		Income Statement Alternative	Possible disclosure	Disclosure is present	True	CONSISTENT	Income Statement Alternative

Again, while these ideas are explained to the reader using rather simple examples in order to make a basic explanation easier; these reporting checklist rules are quite sophisticated in nature. You can see a working prototype of Microsoft's 10-K financial report which was submitted to the SEC using a commercially available tool provided by XBRL Cloud that leverages these reporting checklist rules[201]:

---

[199] *Automating Accounting and Reporting Checklists*,
http://xbrl.squarespace.com/journal/2016/5/5/automating-accounting-and-reporting-checklists.html
[200] Reporting checklist for Proof, machine-readable,
http://xbrlsite.azurewebsites.net/2020/intermediate/proof/dr-rules-def.xml
[201] *Combined Disclosure Mechanics and Reporting Checklist provided by XBRL Cloud*,
http://xbrlsite.azurewebsites.net/2017/Prototypes/Microsoft2017/Disclosure%20Mechanics%20and%20Reporting%20Checklist.html

Disclosure Mechanics and Reporting Checklist

| Entity Registrant Name: | MICROSOFT CORPORATION (SEC Filing Page) | | | | | Document Type: | | 10-K | | | |
| CIK: | 0000789019 | | | | | Fiscal Year / Period: | | 2017 / FY | | | |

Disclosures Found: 48 of 70 (69%)   Disclosures Consistent: 68 of 70 (97%) | Disclosures Inconsistent: 2 of 70 (3%)

Show: ⦿All ◯Only Consistencies ◯Only Inconsistencies ◯Only Reported ◯Only Not Reported   ☐ Show Level 1 Note And Policy Concept Columns

#	Disclosure	Category	Level	Pattern	Applicable	Found	Disclosure Consistent	Representation Concept [TEXT BLOCK]	Representation Concept [DETAIL]	Checklist Category	Reason
1	Document Information [Hierarchy]	DOCUMENT	Level4Detail	HIERARCHY	True	True	CONSISTENT	NOT-EXPECTED	Document Fiscal Period Focus	Required disclosure	Disclosure always required
2	Document and Entity Information [Hierarchy]	DOCUMENT	Level4Detail	HIERARCHY	False	True	CONSISTENT	NOT-EXPECTED	Entity Registrant Name	Alternative representation	Not necessary, satisfied by Document Information [Hierarchy] disclosure
3	Entity Information, by Legal Entity [Hierarchy]	DOCUMENT	Level4Detail	HIERARCHY	True	True	CONSISTENT	NOT-EXPECTED	Entity Registrant Name	Required disclosure	Disclosure always required
4	Document and Entity Information [Hierarchy]	DOCUMENT	Level4Detail	HIERARCHY	False	True	CONSISTENT	NOT-EXPECTED	Entity Registrant Name	Alternative representation	Not necessary, satisfied by Entity Information, by Legal Entity [Hierarchy] disclosure
5	Balance Sheet	STATEMENT	Level4Detail	COMPONENT	True	True	CONSISTENT	NOT-EXPECTED	NOT-EXPECTED	Required	Disclosure always required, satisfied by Assets [Roll Up] and Liabilities

7.6. *Some Final Thoughts*

We began this discussion talking about getting computers to do work and how you use rules to control that work in order to effectively automate processes. For automation to be effective, the results of processing an XBRL-based financial report should be deterministic. That is the only way two parties can agree on what information is being conveyed by such reports effectively.

The new medium for exchanging complex financial information, XBRL, will undoubtedly contribute to reducing the monotonous, repetitive, mechanical tasks and processes accountants need to perform which will free these professionals up to better add value in the information age.

To further your understanding, I would encourage you to work through the analysis of the **Microsoft** 2017 10-K financial report which ties the ideas and examples used in this document to a real world XBRL-based financial report[202].

The examples provided here may seem too simple. They are not too simple; they communicate only what is essential. Consider this more complex example from an actual Microsoft 10-K report[203]:

Debt Instrument [Line Items]	Notes 0.875 Percent Due November 15th 2017	Notes 1.0 Percent Due May 1st 2018	Notes 1.30 Percent Due November 3rd 2018	Notes 1.625 Percent Due December 6th 2018	Notes 4.2 Percent Due June 1st 2019	Notes 1.1 Percent Due August 8th 2019	Notes 0.500 Percent Due November 1st 2019	Notes 1.850 Percent Due February 6th 2020	Notes 1.850 Percent Due February 12th 2020	Notes 3.0 Percent Due October 1st 2020	Notes 2.0 Percent Due November 3rd 2020	Notes 4.0 Percent Due February 8th 2021
Debt Instrument [Set]												
Debt maturity date	2017-11-15	2018-05-01	2018-11-03	2018-12-06	2019-06-01	2019-08-08	2019-11-01	2020-02-06	2020-02-12	2020-10-01	2020-11-03	2021-02-08
Stated Interest Rate	0.875%	1.00%	1.30%	1.625%	4.20%	1.10%	0.50%	1.85%	1.85%	3.00%	2.00%	4.00%
Effective Interest Rate	1.084%	1.106%	1.396%	1.824%	4.379%	1.203%	0.50%	1.952%	1.935%	3.137%	2.093%	4.082%
Face Value	600,000,000	450,000,000	1,750,000,000	1,250,000,000	450,000,000	2,500,000,000	1,500,000,000	1,500,000,000	1,000,000,000	1,000,000,000	2,000,000,000	2,250,000,000

Adding more dimensions or adding more members to a dimension does not get you any closer to testing the essence of XBRL-based financial reporting. Sure, you can test to see how well such reports scale. But that is a completely different subject. Here, we are focused on effectiveness.

---

[202] Microsoft XBRL-based Financial Report Analysis, http://xbrl.squarespace.com/journal/2020/4/13/microsoft-xbrl-based-report-analysis.html

[203] Microsoft, Complex Example 1, http://xbrlsite.azurewebsites.net/2020/master/complex1/reference-implementation/evidence-package/contents/index.html#Rendering-DisclosureLongTermDebtDetail-base_DebtInstrumentTable.html
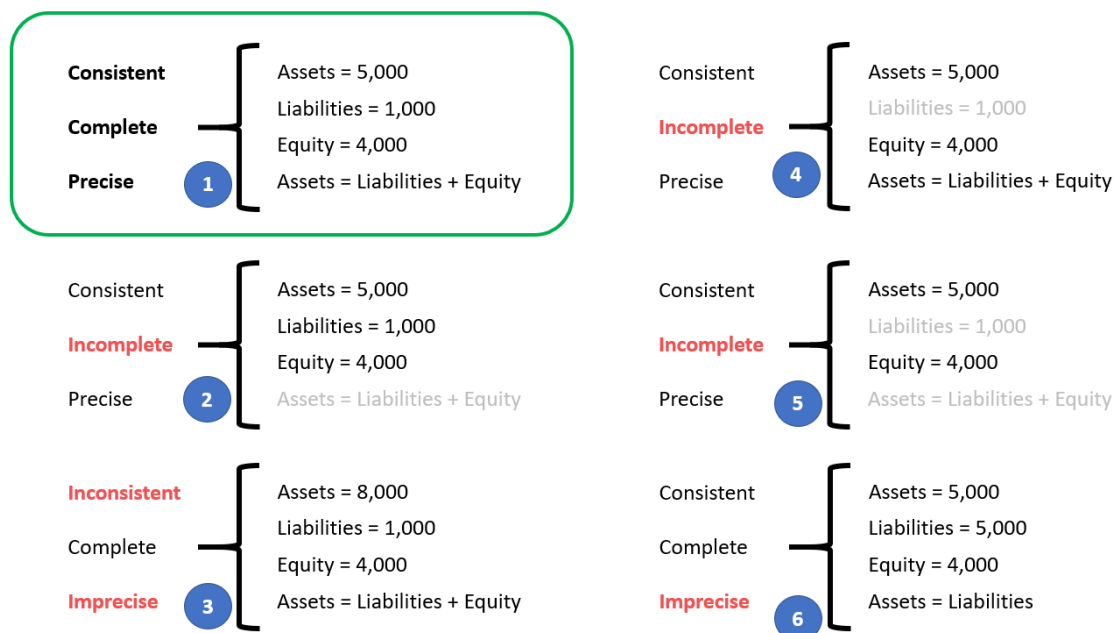
8. Understanding Errors that Can Occur

An excellent way of making sure things are right is by studying what can go wrong. This helps you understand the impediments to creating a properly functioning logical system.

The following is a comparison of 9 states of the simple financial report logical system, the accounting equation[204]. The point of using such a simple financial report logical system is to explain specific things that can go wrong so that a reader can understand why each of the categories of rules are necessary to control a process to make sure things are in fact right. These 9 states can occur in any financial report with one fragment, two fragments, or 194 fragments like the Microsoft 10 K. Again, *Mastering XBRL-based Digital Financial Reports*[205] walks you through small and simple to large and complex. For more information about these impediments, please watch the video playlist Understanding the Financial Report Logical System[206], in particular the three videos related to impediments.
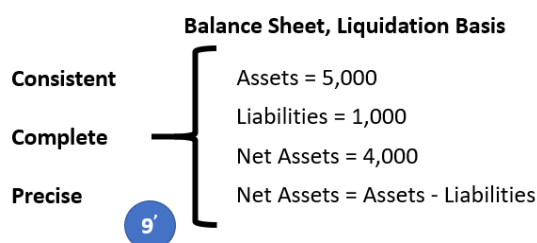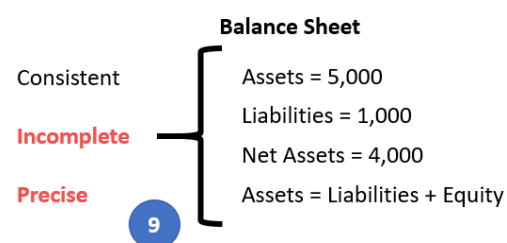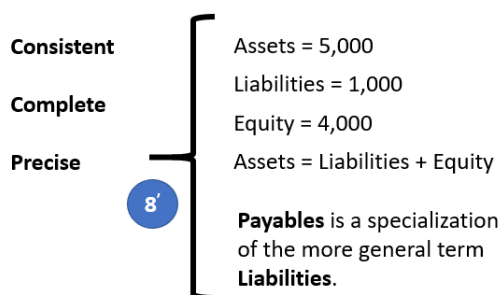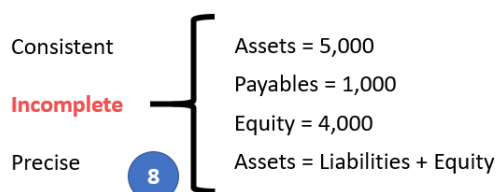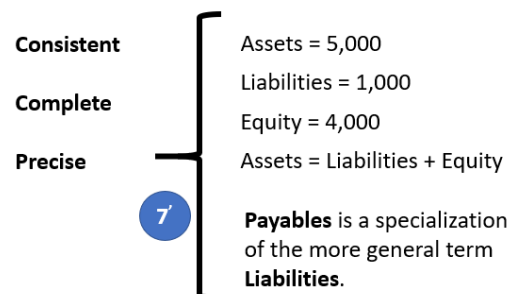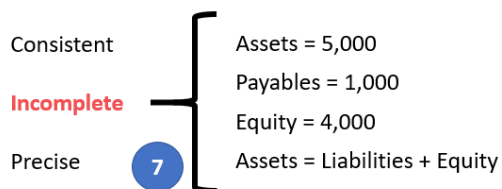
Here is a summary of all nine states with the first state outlined in green being the only properly functioning logical system proven to be complete, consistent, and precise:

---

[204] Accounting equation, http://xbrlsite.azurewebsites.net/2020/master/ae/index.html

[205] Mastering XBRL-based Digital Financial Reports, http://xbrlsite.azurewebsites.net/2020/master/

[206] Understanding the Financial Report Logical System, https://www.youtube.com/playlist?list=PLqMZRUzQ64B7EWamzDP-WaYbS_W0RL9nt

Consistent

Incomplete

Precise **7**

Assets = 5,000
Payables = 1,000
Equity = 4,000
Assets = Liabilities + Equity

Consistent

Complete

Precise **7′**

Assets = 5,000
Liabilities = 1,000
Equity = 4,000
Assets = Liabilities + Equity

Payables is a specialization of the more general term **Liabilities**.

Consistent

Incomplete

Precise **8**

Assets = 5,000
Payables = 1,000
Equity = 4,000
Assets = Liabilities + Equity

Consistent

Complete

Precise **8′**

Assets = 5,000
Liabilities = 1,000
Equity = 4,000
Assets = Liabilities + Equity

Payables is a specialization of the more general term **Liabilities**.

Balance Sheet

Consistent

Incomplete

Precise **9**

Assets = 5,000
Liabilities = 1,000
Net Assets = 4,000
Assets = Liabilities + Equity

Balance Sheet, Liquidation Basis

Consistent

Complete

Precise **9′**

Assets = 5,000
Liabilities = 1,000
Net Assets = 4,000
Net Assets = Assets - Liabilities

In the following sections I want to make some adjustments to the logical system which make the logical system either inconsistent, incomplete, or imprecise and explain why the system is then not a properly functioning logical system. I made videos that explain each of these impediments to a properly functioning logical system which you can see in this video playlist, *Understanding the Financial Report Logical System*[207].

Before we get to the improperly functioning logical systems, let's take one final look at the properly functioning logical system so that you can use that as a baseline for comparing and contrasting the properly functioning and improperly functioning logical systems so that you can understand they sorts of errors that can occur.

SUMMARY:

- **State 1**: A report can be provably properly functioning per a set of rules. The State 1 report is an example of a properly functioning financial report. **All the logical statements are provided for per a set of machine-readable rules, the statements are consistent with one another, and the information is precise per the area of knowledge (in this case the accounting equation)**. The Pacioli logic engine verifies all this.

- **State 2**: If you REMOVE a logical statement, such as if you REMOVE the rule "Assets = Liabilities + Equity"; and three facts are reported; a machine-based

---

[207] Understanding the Financial Report Logical System,
https://www.youtube.com/playlist?list=PLqMZRUzQ64B7EWamzDP-WaYbS_W0RL9nt

process can have NO IDEA whether those three facts are CORRECT or INCORRECT using automated processes because the RULE IS MISSING. NO MACHINE-READABLE RULE, MACHINE CANNOT VERIFY.

- **State 3**: If you intentionally put information in the report to simulate an error; Pacioli DETECTS that error and a human can SEE that from the verification results. But for inconsistencies to be detected, the machine-readable rules must exist.

- **State 4**: If a line item is not reported, this causes increased complexity in processing reports. For example, if you leave out the line item "Liabilities" BUT you don't provide a machine-readable derivation rule, there is NO WAY a computer can UNDERSTAND THE information.

- **State 4'**: But as State 4' shows, if you DO provide the derivation rules, the system will still be able to automate effectively because the missing information can be derived. It just calls for additional work (i.e. creating the derivation rule BECAUSE not reporting certain line items is allowed). Don't what to cause additional work? Don't want to cause additional RISK of misunderstanding? Then DON'T ALLOW unreported high-level line items.

- **State 5**: If a line item is NOT reported AND the consistency rule is not provided; THEN there is no way a machine-based process can effectively use the report. Remember, computers are INCREDIBLY DUMB. They need to be led by the hand; that is what machine-readable rules do.

- **State 6**: It is possible to represent a report, get 100% of the verification checks to be GREEN; but the report is STILL WRONG. If (a) a FACT is wrong and (b) if a RULE is wrong; the two can work together to make a report seem CORRECT. However, this situation can be detected by using high-level crosschecks of the continuity of a report.

- **State 7**: If a report uses an extension concept to report a fact, the MACHINE will not understand how to process the extension information, so humans MUST step in to sort things out. As such, a process cannot be automated.

- **State 7'**: "Anchoring" helps OVERCOME State 7; the machine-readable anchoring information enables a process (a) to be automated and (b) the use of extensions to make the system more flexible which maximizes system "information richness".

- **State 8**: The EXACT SAME PROBLEM that anchoring solves with extension concepts exists with BASE TAXONOMY CONCEPTS!!! Again, computers are dumb, dumb, dumb. If anchoring is not also used in the base taxonomy, then you have EXACTLY the same problem you have with extensions.

- **State 9**: The EXACT SAME PROBLEM that anchoring solves exists with DISCLOSURES (i.e. SETS of concepts). Whether a disclosure (structure) is in a report model or base taxonomy model; if an automated process cannot sort out what that structure/disclosure is; then it WILL NOT KNOW WHAT TO DO WITH IT. Anchoring applies to structures/disclosures as it does to concepts, report or base models.

- **State 10**: The report models need to be represented logically. What does "indentation" of a concept mean? Whatever you thing it means, it means something else to someone else. Every Tom, Dick, and Harry has their own

personal interpretation. Don't use "indentation" (parent-child associations) to carry ambiguous meaning. Remember, computers are REALLY dumb.

- **State 11**: Fundamentally, the "payload" of information must be provided by a standard technical syntax. It an improper XBRL technical syntax is used, software may not work correctly. XBRL technical syntax tends to be pretty good because XBRL International publishes a conformance suite which can be used to test software to see if the software is detecting mistakes correctly in XBRL technical syntax.

Next, we will elaborate on the details of each state a bit more.

8.1. State 1: Properly Functioning Logical System

For completeness, I want to start by mentioning again our properly functioning logical system which is consistent, complete, and precise. It can be helpful to contrast other states to this state to understand the difference between properly functioning logical systems and improperly functioning systems.

Again, this is considered a properly functioning logical system because (a) all the statements within the system are **consistent**; (b) the set of statements that describe the system is **complete**; and (c) the information conveyed by the system is **precise** in its representation of reality. Further, we are formally declaring this "reality"[208] to be our base understanding.

Also, we need to be explicit. We defined three terms "Assets", "Liabilities", and "Equity".

Now, you may know what those three terms are; but a computer does not. You have to define what you work with relative to something that you know. Imagine our system defines four terms, "fac:Assets", "fac:Liabilities", "fac:Equity", and "fac:LiabilitiesAndEquity"[209]. You understand your system but you have to map every external system into your system[210]. Your internal system understands more that the accounting equation system (i.e. you have LiabilitiesAndEquity). You have to be able to compute that value based on some other system's information[211]. It is perfectly reasonable for our system to create a concept LiabilitiesAndEquity and compute that value even though the accounting equation logical system does not have that explicit value.

---

[208] YouTube, *Reality*, https://youtu.be/eq2Jw6waaCI

[209] Fundamental accounting concepts, http://xbrlsite.azurewebsites.net/2020/core/master-ae/fac.xsd

[210] Mapping from accounting equation to fundamental accounting concepts in our system, http://xbrlsite.azurewebsites.net/2020/core/master-ae/fac-mapping-definition.xml

[211] XBRL Formula to derive the value for LiabilitiesAndEquity, http://xbrlsite.azurewebsites.net/2020/core/master-ae/fac-ImputeRule-LiabilitiesAndEquity-formula.xml

The point is that different economic entities have different models; but all models of a financial reporting scheme are reconcilable from/to one another in some manner[212].

8.2. State 2: Incomplete Coverage by Rules

The logical system #2 below is intended to show exactly the same information as our #1 properly functioning logical system, except that #2 leaves out the rule "Assets = Liabilities and Equity" which is showed as grayed out (i.e. because it is assumed to be missing from the logical system.

Coverage is a measure of how well you do or can represent a domain of information within a logical system. "Do" is about using the tools you have correctly and effectively. "Can" is about the capabilities of the tools you are using to represent the rule.

For example, if your logical system neglects to include the rule "Assets = Liabilities + Equity" or if your tools don't provide the capabilities to allow you to represent that rule; then there is the possibility that the facts being represented to be represented incorrectly and the system will not detect the inconsistency. As such, that logical system has **incomplete coverage**.
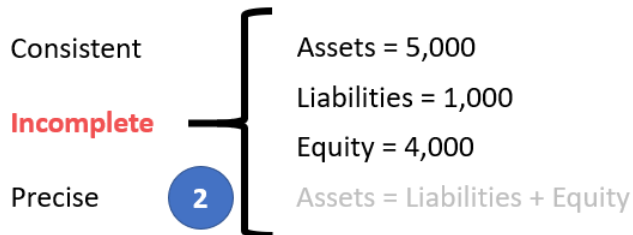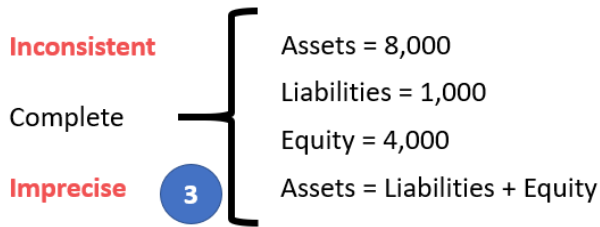

While this specific state #2 does have the Assets, Liabilities, and Equity facts consistent with the absent rule; the system is still incomplete because the coverage can be improved by adding the missing rule. If that missing rule is added, then the logical system can be considered complete again.

8.3. State 3: Inconsistent and Imprecise

All the statements in the system must be consistent for the logical system to be considered properly functioning. If statements are inconsistent, the logical system is not is not properly functioning. In this system #3, the values for Assets, Liabilities, and Equity are inconsistent with the rule "Assets = Liabilities + Equity". From looking at the information provided, it is impossible to know exactly which of the three facts are incorrect; it is only possible to understand that the statements made within the logical system is inconsistent. It could be the case that the rule is incorrect.

---

[212] Charles Hoffman, CPA, *Special Theory of Machine-based Automated Communication of Semantic Information of Financial Statements*, http://xbrl.squarespace.com/journal/2019/12/30/special-theory-of-machine-based-automated-communication-of-s.html

Inconsistent

Complete

Imprecise (3)

Assets = 8,000
Liabilities = 1,000
Equity = 4,000
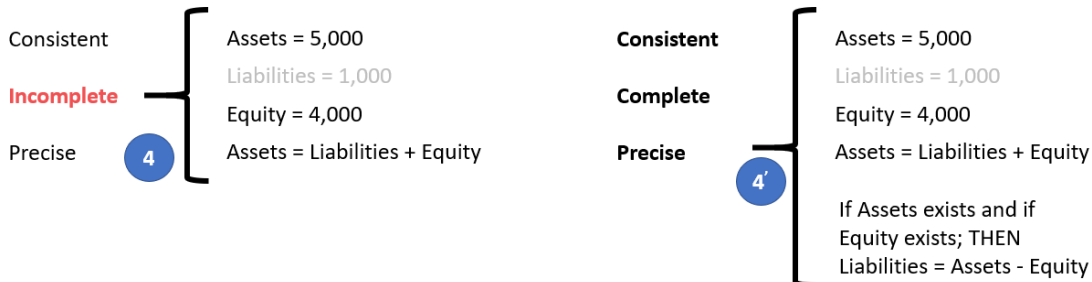Assets = Liabilities + Equity

However, given that we know from state #1 that the value for Assets is 5,000 and not 8,000; the facts in this system is imprecise because the fact for Assets does not reflect reality.

8.4. State 4: Unreported Facts

In state #4, the situation is that the economic entity representing information in their report neglected to include the fact for Liabilities. Whether it is the case that a fact can, or cannot, be left unreported is a decision that can be made by the stakeholders of the system.

If it is the case that it is decided that the fact "Liabilities" can be omitted if both Assets and Equity are reported; then you must provide a rule to derive the value of Liabilities when that fact is not reported. Below you see that the system has been adjusted in state #4' to add the rule "IF Assets exists and if Equity exists; THEN Liabilities = Assets - Equity"[213]. *(NOTE that this rule should actually be "IF Assets exists and if Equity exists and if not(exists) Liabilities; THEN Liabilities = Assets – Equity")*

Consistent

Incomplete

Precise (4)

Assets = 5,000
Liabilities = 1,000
Equity = 4,000
Assets = Liabilities + Equity

Consistent

Complete

Precise (4')

Assets = 5,000
Liabilities = 1,000
Equity = 4,000
Assets = Liabilities + Equity

If Assets exists and if Equity exists; THEN Liabilities = Assets - Equity

If it were likewise true that either Assets[214] or Equity[215] could also be left unreported, similarly derivation rules could be created for each of those facts. Note that XBRL Formula chaining[216] can be used to physically derive unreported facts if any one of these three facts remain unreported. Note that it is impossible to derive missing information if any two of the facts remain unreported. Adding the derivation rule makes the system complete.

---

[213] Here is the impute or derivation rule that would be added to the accounting equation logical system for this situation, http://xbrlsite.azurewebsites.net/2020/core/master-ae/ImputeRule-Key-1-Code-BS-Impute-01-formula.xml

[214] XBRL Formula rule for deriving Assets, http://xbrlsite.azurewebsites.net/2020/core/master-ae/ImputeRule-Key-3-Code-BS-Impute-03-formula.xml

[215] XBRL Formula rule for deriving Equity, http://xbrlsite.azurewebsites.net/2020/core/master-ae/ImputeRule-Key-2-Code-BS-Impute-02-formula.xml

[216] Deriving Facts Using XBRL Formula Chaining (Example), http://xbrl.squarespace.com/journal/2019/4/24/deriving-information-using-xbrl-formula-chaining-example.html

Allowing certain line items of a report to go unreported specifies the need to create rules to derive missing information. Or saying this another way, omitting the possibility of unreported facts negates the need for creating derivation rules.

A second downside of allowing unreported facts is that you lose the parity check or cross check if facts can go unreported. Said another way, it would be considered best practice to not leave important high-level financial report line items to go unreported.

8.5. State 5: Incomplete

Similar to state #4, in state #5 the logical system is incomplete because both (a) the fact Liabilities is unreported and also (b) the consistency rule "Assets = Liabilities + Equity" is missing from the logical system. Because both a fact and the rule are missing from the logical system, it would be impossible to deduce the value of Liabilities in this case. There is not enough information in the logical system to allow Liabilities to be derived. At a minimum, a consistency crosscheck rule[217] plus the derivation rule to impute Liabilities[218] would be necessary.
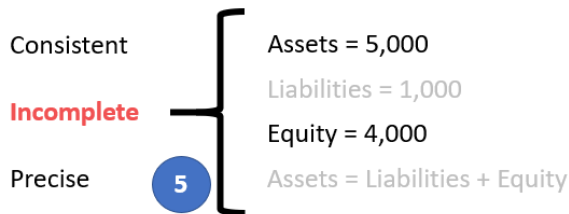
Again, consistent with state #4; Assets and Equity would require similar rules and there is no parity check of reported information.
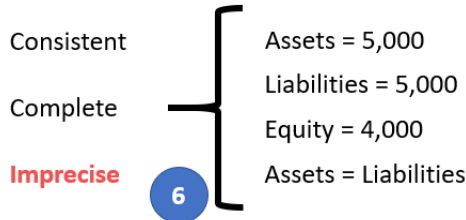
8.6. State 6: Imprecise

A logical system is a true and fair representation of some agreed upon realism. **Precision** is a measure of how precisely you do or can represent the information of a domain within a logical theory. The reality that we formalized in state #1 indicates that "Assets = Liabilities + Equity". Yet, in the state #6 example, the rule "Assets = Liabilities" was provided. Further, the values of Assets and Liabilities are, in fact, consistent with the rule that has been provided.

Remember that in state #1 we formalized our truth to be that "Assets = Liabilities + Equity". As such, this logical system can be described as being imprecise. To make this logical system precise, all that needs to be done is to fix the rule.

---

[217] XBRL Formula consistency crosscheck rule Assets = Liabilities + Equity, http://xbrlsite.azurewebsites.net/2020/core/master-ae/Consistency-5-Code-BS01-formula.xml
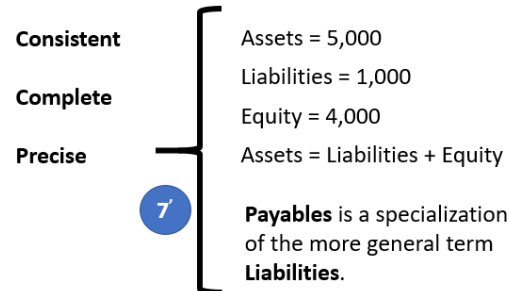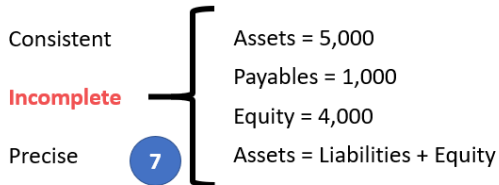
[218] XBRL Formula derivation rule to impute Liabilities, http://xbrlsite.azurewebsites.net/2020/core/master-ae/ImputeRule-Key-1-Code-BS-Impute-01-formula.xml

Consistent

Complete

Imprecise ⑥

Assets = 5,000

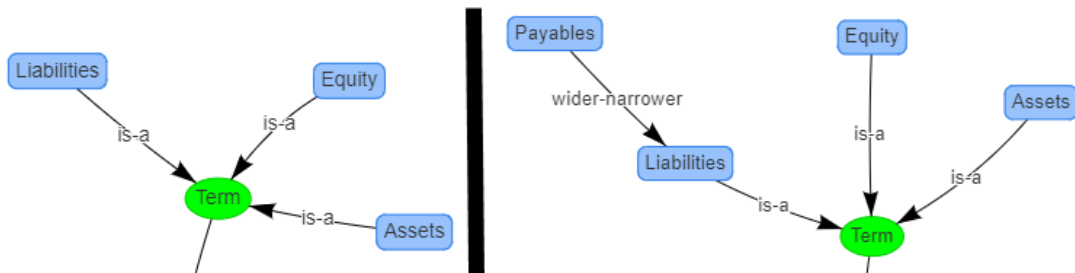Liabilities = 5,000

Equity = 4,000

Assets = Liabilities

8.7. State 7: Extension Concept

In state #7 on the left, what we are trying to convey is that the economic entity reported the fact for Liabilities using the extension concept "Payables" that it had created. If a fact is represented using an extension concept created by a reporting entity; then a "general-special" or "wider-narrower" or "class-equivalentClass" association must be created to indicate to software applications of the relationship so that information can be used correctly. State #7' on the right, the rule "Payables is a specialization of the more general term Liabilities" has been added to the logical system which allows the system to operate effectively[219].

Consistent

Incomplete

Precise ⑦

Assets = 5,000

Payables = 1,000

Equity = 4,000

Assets = Liabilities + Equity

Consistent

Complete

Precise

⑦'

Assets = 5,000

Liabilities = 1,000

Equity = 4,000

Assets = Liabilities + Equity

Payables is a specialization of the more general term **Liabilities**.

And so, the graphic below shows a fragment of the knowledge graph on page 9 above before and after the information that "Payables is a specialization of the more general term Liabilities," was added. On the left you see State 7, the taxonomy before the information was added and on the right you see "Payable" being added as an extension concept indicating that there is a "wider-narrower" relationship between Payables and Liabilities. Therefore, a machine based process can utilize the information per State 7' because the process understands Liabilities in the taxonomy, understands the "wider-narrower" relationship therefore knowing that "Payables" is a type of Liability.


---

[219] XBRL Definition relations showing example of a mapping rule, http://xbrlsite.azurewebsites.net/2020/core/master-ae/fac-mapping-definition.xml

## 8.8. State 8: Base Taxonomy Wider/Narrower Concept Use

State #8 on the left below is similar to state #7 in that a different concept is used to report a fact; but while state #7 focuses on using an extension concept; state #8 points out that using a wider or narrower base taxonomy concept gives exactly the same result.

Now, our base state #1 does not have the concept "Payables"; but let's assume for a moment that it does have the concept "Payables". Also suppose that there was no information in the base logical system indicating the relationship between "Payables" and any other concept. If a fact is represented using a BASE TAXONOMY CONCEPT by a reporting entity; then a "general-special" or "wider-narrower" or "class-equivalentClass" association must exist in that base taxonomy to indicate that some concept is a permissible alternative for some other concept.

State #8' on the right adds the rule "Payables is a specialization of the more general term Liabilities"[220].
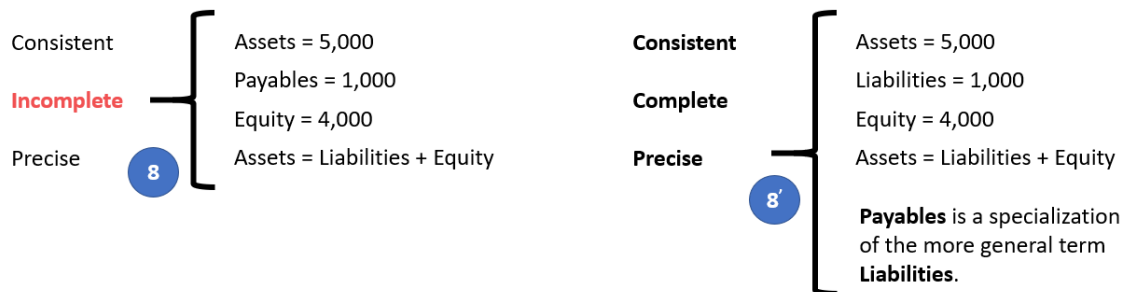
And so, the graphic below shows a fragment of the knowledge graph on page 9 above before and after the information that "Payables is a specialization of the more general term Liabilities," was added. On the left you see State 8, the base taxonomy before the information was added and on the right you see "Payable" being added as an extension concept indicating that there is a "wider-narrower" relationship between Payables and Liabilities. Therefore, a machine based process can utilize the information per State 8' because the process understands Liabilities in the base taxonomy, understands the "wider-narrower" relationship therefore knowing that "Payables" is a type of Liability.
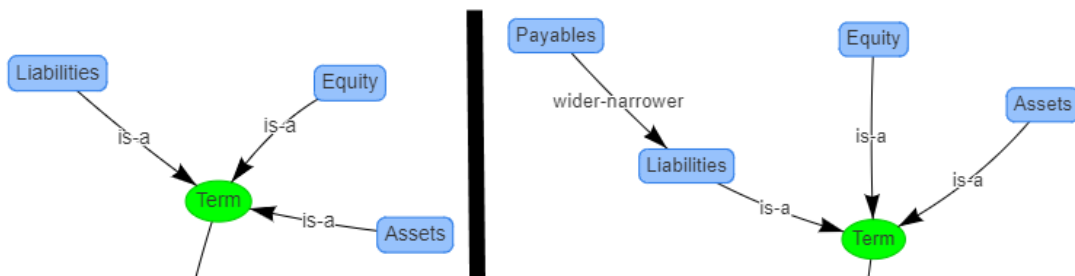

8.9. State 9: Defining a Completely New Structure

State #9 below on the left focuses on the structure as contrast all the prior examples which focused on the terms and rules. If a new structure is created, the new structure

---

[220] XBRL Definition relations showing example of a mapping rule,
http://xbrlsite.azurewebsites.net/2020/core/master-ae/fac-mapping-definition.xml

must be referenced to the base taxonomy and the new structure needs to be explained using machine-readable rules[221]. Even base taxonomy structures need to be defined in order to be referred to[222]. When you say "Balance Sheet" you know what that means. But a machine does not know.

A base taxonomy should (a) provide all necessary structures separately, not intermingle different models in the same set of associations and (b) define what each structure must look like. Remember, computers are like babies and need to be led by the hand in order to understand the details you need them to understand.
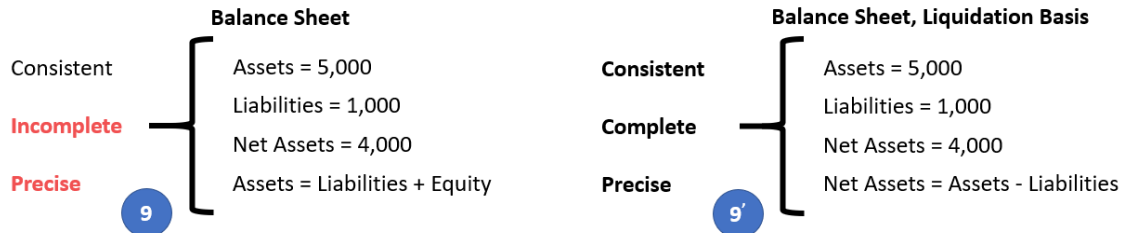
Finally, in our case we have only one disclosure, the Balance Sheet. In our case, the Balance Sheet is always required to be reported per this logical system. As such, that rule is stated in a machine-readable reporting checklist[223]. Other logical systems with more disclosures will have more rules relating to when a disclosure is required to be provided in a report.

Similar to how "Payables" was added as an extension of the terms in the logical system; we can extend the structures to include a "Liquidation Basis Balance Sheet" structure which is a specialization of a Balance Sheet:
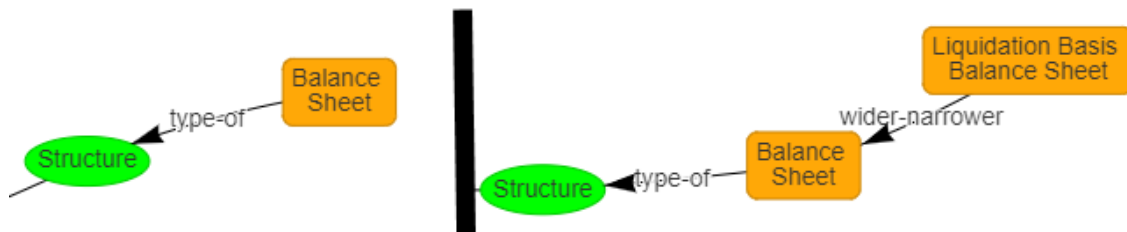
And such, an automated process will be able to understand the new structure because it is related to an existing structure. Other structures could be added and only identified as a type of structure.

9.   Examining Errors in Actual Submitted Reports

As explained, a very good way to understand how to create reports correctly is to examine errors in reports that others have created. Here are three documents that provide well documented examples of undisputed errors:

---

[221] XBRL Definition relations used to represent structure rules,
http://xbrlsite.azurewebsites.net/2020/core/master-ae/dm-1355-rules-def.xml

[222] XBRL taxonomy schema used to define "Balance Sheet",
http://xbrlsite.azurewebsites.net/2020/core/master-ae/disclosures.xsd

[223] XBRL Definition relations used to represent a reporting checklist or disclosure rules,
http://xbrlsite.azurewebsites.net/2020/core/master-ae/reporting-checklist-rules-def.xml

- Issues in XBRL-based Digital Financial Reports[224]
- More Issues in XBRL-based Digital Financial Reports[225]
- Accounting errors[226]

Finally, this blog post[227] (see toward the bottom) provides 24 sets of well documented errors in XBRL-based reports.

Note that all of these errors were discovered using automated processes which leverage this method. If reports were verified using this method prior to submitting the report to a regulator, all such errors could have been detected and corrected. The result would be higher quality XBRL-based reports.

10. Best Practices for Preventing Errors

A **best practice** is a method or technique that has been generally accepted as superior to any other known alternatives because it produces results that are superior to those achieved by other means or because it has become a standard way of doing things.

Best practices (or good practices) are techniques that have produced outstanding results in other situations, inside or outside of a particular organization and which can be validated, codified, and shared with others and recommended as models to follow[228].

When one understands these dynamics of an XBRL-based report when a reporting economic entity can "reshape" or "alter" or otherwise modify their report then one will understand why some method[229] for controlling such modifications is necessary. This control mechanism contributes to making sure that processes yield high-quality financial information that is provided by such reports.

The method I use is the *Seattle Method*[230].

11. Rules of Thumb

The following is a summary of the rules of thumb for creating XBRL-based digital financial reports. The details of these rules of thumb can be found in the document Rules of Thumb[231] which provides a very detailed explanation of each rule, reasoning

---

[224] Issues in XBRL-based Digital Financial Reports,
http://xbrlsite.azurewebsites.net/2020/master/100IssuesOfPublicCompanyReports.pdf
[225] More issues in XBRL-based Digital Financial Reports,
http://xbrlsite.azurewebsites.net/2020/master/About50MoreIssuesOfPublicCompanyReports.pdf
[226] Accounting Errors,
http://xbrlsite.azurewebsites.net/2018/Library/AccountingErrorsFoundDuringValidation.pdf
[227] High Quality Examples of Errors in XBRL-based Financial Reports,
http://xbrl.squarespace.com/journal/2017/4/29/high-quality-examples-of-errors-in-xbrl-based-financial-repo.html
[228] European Guide to good Practice in Knowledge Management - Part 5: KM Terminology, page 3 (PDF page 9), http://arielsheen.com/wp-content/uploads/2019/10/CEN-CWA14924-05-2004-Mar.pdf#page=9
[229] *Understanding Method (Abridged)*,
http://xbrlsite.azurewebsites.net/2020/Library/UnderstandingMethod_Abridged.pdf
[230] Charles Hoffman, CPA, Seattle Method, http://xbrlsite.com/seattlemethod/SeattleMethod.pdf
[231] *Rules of Thumb*, http://www.xbrlsite.com/mastering/Part04_Chapter07.G4_RulesOfThumb.pdf

for the rule, and examples that help you understand the rules. The rules of thumb help you tie some loose ends together in your mind.

(NOTE: This is under construction; please check back on the Rules of Thumb document periodically)

1. Always use explicit, closed hypercubes.

2. Explicitly hypercubes provided should always be unique.

3. Select dimensions carefully.

4. Assume facts need to be uniquely identifiable across all reports in a system.

5. All concepts defined should be included in at least one hypercube.

6. Dimension defaults specified by a base financial reporting scheme taxonomy should never change.

7. Core information pattern models should be understood, respected, and always followed.

8. Understand and respect logical relations between report element categories.

9. Properly separate information blocks.

10. Accounting concepts should never be included in dimensions.

11. Always represent roll forward computations as line items.

12. Never define facts in multiple ways (needs fixing)

13. Dimensions used in the representation of a fact do not represent properties.

14. Dependent dimensions should not be used.

15. Use type members to define an indefinite series of members.

16. Extensible enumerations should be used to represent facts associated with dimensions with a single member.

17. Properties (a.k.a. traits, qualities, attributes) should be represented using XBRL definition relations using the trait arcrole.

18. Atemporal concepts should be durations with a period type of forever.

19. Facts representing specific transaction details should have period type value of forever.

20. Report element names and properties must never be changed in a base financial reporting scheme XBRL taxonomy.

21. Transactions from beginning of time to some end date are considered instants (not durations).

12. Next Steps

This *Showcase of Reports*[232] helps one understand the current capabilities of XBRL-based digital Financial Reporting. The examples also help you understand important nuances and subtleties of such reports. Finally, the examples can be used to help you

---

[232] Showcase of Reports,
http://www.xbrlsite.com/mastering/Part04_Chapter07.1_ShowcaseOfReports.pdf

separate good software from not so good software by trying the examples in different software applications.

Be aware of the information provided in the two blog posts *PLATINUM XBRL-based Financial Report Examples*[233]and *PLATINUM Business Use Cases, Test Cases, Conformance Suite*[234]*.*

To understand XBRL-based financial reporting in more detail please have a look at *Mastering XBRL-based Digital Financial Reports*[235]. That document is a summary and synthesis of all the information from my blog[236].

No one knows for certain how the future will unfold. I have provided my predictions based on the empirical evidence that I have collected. An enhanced/improved version of BI software is likely[237], new modern tools for professional accountants will be created[238], modern logic oriented semantic spreadsheets[239], semantic oriented accounting and audit working papers and schedules[240].

In the future, my best information will be provided in the documents that make up *Mastering XBRL-based Digital Financial Reporting*[241]. Watch for updates there.

In time, the answer will be revealed.

---

[233] PLATINUM XBRL-based Financial Report Examples, https://digitalfinancialreporting.blogspot.com/2023/07/platinum-xbrl-based-financial-report.html

[234] PLATINUM Business Use Cases, Test Cases, Conformance Suite, https://digitalfinancialreporting.blogspot.com/2023/07/platinum-business-use-cases-test-cases.html

[235] *Mastering XBRL-based Digital Financial Reports*, http://xbrl.squarespace.com/mastering-xbrl/

[236] XBRL Blog Archive, http://xbrl.squarespace.com/blog-archive/

[237] *XBRL+SBRM=Improved BI*, https://digitalfinancialreporting.blogspot.com/2023/08/xbrlsbrm-improved-bi.html

[238] Creating New Tool for Accounting and Audit, https://digitalfinancialreporting.blogspot.com/2023/08/creating-new-tool-for-accounting-and.html

[239] Modern Spreadsheet, https://digitalfinancialreporting.blogspot.com/2023/05/modern-spreadsheet.html

[240] Semantic Accounting and Audit Working Papers, https://digitalfinancialreporting.blogspot.com/2023/05/semantic-accounting-and-auditing.html

[241] *Mastering XBRL-based Digital Financial Reporting*, http://xbrl.squarespace.com/mastering-xbrl/