1. Digging into Slots, Blocks and the Mechanics of a Business Report

This section digs into the important ideas of Blocks and Slots more to help solidify your understanding of these important ideas.

If you go to a paper-based financial report you will not find any mention of a "block" or a "slot". The Block and the Slot are useful ideas when trying to get a computer to understand digital financial reports. These notions help you think about and discuss these reports.

Think of an electronic spreadsheet. You have the idea of "row" and "column" and "cell" to help you interact with the spreadsheet. You can insert a row, delete a column. But you cannot delete a cell. You delete cells by deleting a row or deleting a column which contains the cell.

This section helps you understand these ideas as they relate to a business report. A financial report is a type of business report. Therefore, financial reports need to follow the rules of business reports. Financial reports add new ideas, building upon the notion of a business report. An SEC-style XBRL-based digital financial report both builds on business reports and further restricts how the XBRL technical syntax is used to construct XBRL-based digital financial reports. These ideas will be covered in subsequent sections. For right now, just realize that these ideas are key to understanding how an XBRL-based digital financial report works. You can think of the digital business report and digital financial report as being different layers of the same conceptual model. Establishing these layers properly increases flexibility and utility of these digital tools.

These models were gleaned by reverse-engineering XBRL-based financial reports submitted to the U.S. Securities and Exchange Commission (SEC) by public companies over a period of about five years. Every such report has these characteristics.

1.1. Overview

By basic mechanics of a financial report we simply mean the basic important real world things that make up a financial report and the basic important logical, mechanical, structural, and mathematical relations between those real world things, the essence of a financial report. This is done so that we can then explain how a financial report works to a machine, such as a computer, so that the machine can help us create and make use of the information contained within financial reports. The machine needs to be able to interact with these report characteristics.

With paper-based reports understanding these mechanical pieces and describing them is not nearly as important because tools used to create such paper-based reports are presentation oriented and have no knowledge of a financial report. Further, these reports are structured for presentation of the information rather than the meaning of information. Humans interact with these software tools used for creating financial reports by interacting with things such as paragraphs, tables, rows, columns, and cells.

1.2. Understanding the notion of patterns

A pattern is a representation model or set of rules which are used to guide. Patterns are important to information technology professionals who build software because they help understand what the software does and build software that is useable by business professionals. Computers can leverage patterns. Patterns are both a communications tool that can help business professionals and IT professionals communicate, functionality templates which can be leveraged to make software easier to create, and a specimen that exemplifies the ideal qualities of something.

Basically, patterns describe.

A pattern is something that recurs. The world is full of patterns and information technology engineers leverage these patterns when trying to get a computer to do something effectively and efficiently for humans. Understanding the patterns which exist can help make both building and using software easier.

Patterns perform two fundamental tasks. First, they enable implementation of functionality at a higher-level. So rather than working with low-level pieces, business professionals can interact with higher-level ideas. This is like working with "Lego blocks" to build something. Second, patterns provide boundaries. Boundaries are necessary in order to make a system work safely and predictably.

Everything in the conceptual model of a digital financial reports is essentially high-level patterns. Only identified patterns are allowed. If a new pattern is identified, that pattern can be added.

1.3. Understanding the notion of slot or opening

Like we said earlier, you never really delete a "cell" in an electronic spreadsheet. To delete a cell, you delete the row or column which contains that cell. You can add/remove a row or column. So rows and columns are slots where things can be added to an electronic spreadsheet. Similarly, digital financial reports have slots.

Think of a form. While a form is finite but inflexible, a financial report is finite but also flexible. The difference between the two can be described using the notion of a "slot" or "opening". A form has no slots or openings. A form only has cells into which information may be placed.

A **slot** is simply the idea of an allotted place in an arrangement where something can be logically and sensibly placed. Slots standardize where a financial report can be edited and where it cannot be edited.

Let's start by looking at the Slots of a Block. Recall that a **Block** is simply a convenient unit of a financial report. A Block is a set of facts that in some way go together. Recall that all the Facts in a Block share the same Concept Arrangement Pattern.

Below is an example of a Block. This Block is a "Roll Up". All the Facts in the Block work together to make up a Disclosure for the components of property, plant, and equipment, net.

Suppose you wanted to add something to a roll up of property, plant and equipment as shown below:

Property, Plant and Equipment, by Component [Line Items]	Period [Axis]	
	2010-12-31	2009-12-31
Property, Plant and Equipment, by Component [Roll Up]		
Land	1,000,000	1,000,000
Machinery and equipment, gross	2,000,000	2,000,000
Furniture and fixtures, gross	6,000,000	6,000,000
Accumulated depreciation	(1,000,000)	(1,000,000)
Property, plant and equipment, net	8,000,000	8,000,000

You cannot add a second total to a roll up as a roll up only has one total. It would not make logical sense to add a second total to a roll up. Having two totals in a roll up is illogical or irrational; even morbid or pathological.

What does make sense is to add another line item which makes up the total of the roll up, somewhere in the list of existing line items. One slot is adding a line item between *Land* and *Machinery and equipment, gross*. For example, "Airplanes" might make logical sense. Another slot is adding a line item before the first item *Land*.

Further, *what* you add to the list is also constrained. For example, what you add needs to be a number as a roll up involves showing how some list of numbers rolls up. You would not add text to a roll up. And it cannot be just any number, it needs to be an "as of" type number (as contrast to a "for the period" number from, say, the income statement). Why? Because all of the other numbers in the list are "as of" some balance sheet date, not "for the period" of some income statement or cash flow statement period.

There is another slot which makes sense in the information above. You can see that there are two existing periods. Adding information for a third or even more periods makes sense. It could also make sense to add an entirely new characteristic such as Geographic Area [Axis] and describe the physical location of the component of property, plant and equipment.

It makes no sense to simply add whatever you want, where ever you want, randomly or arbitrarily to the roll up. While every slot or opening where it makes sense to add information to the existing information above has not explicitly been pointed out, the set of examples provide should help you understand the notion of a slot.

1.4. Understanding the notion of block or report fragment

A **Block** is a unit of a digital financial report that is created as a matter of pure convenience. Having the notion of Block simply helps to be able to talk about a specific aspect of a digital financial report.

The best way to understand what a Block is and why having the notion of a Block is useful is to look at a few examples. Again, recall that a Block is a set of facts that go together and share the same Concept Arrangement Pattern.

Consider, again, the following financial report disclosure which we looked at above to explain the notion of a Slot:

Property, Plant and Equipment, by Component [Line Items]	Period [Axis]	
	2010-12-31	2009-12-31
Property, Plant and Equipment, by Component [Roll Up]		
Land	1,000,000	1,000,000
Machinery and equipment, gross	2,000,000	2,000,000
Furniture and fixtures, gross	6,000,000	6,000,000
Accumulated depreciation	(1,000,000)	(1,000,000)
Property, plant and equipment, net	8,000,000	8,000,000

As we pointed out, that disclosure is a roll up of the components that make up property, plant, and equipment, net. The Facts of that disclosure above work together to describe the components of property, plant, and equipment, net. Every Fact is part of that disclosure; the parts form a set of facts, and the set of facts work together to make up the disclosure. There is a one-to-one correlation between the concept arrangement pattern (i.e. roll up in this case) and the Block or set of facts which make up the disclosure.

This will become clearer when we point out another Block that has a different concept arrangement pattern.

The fragment of a financial report provides a different disclosure. This disclosure is a roll forward of the product liability accrual of some the economic entity providing this disclosure:

Schedule of Accrued Liabilities [Line Items]	Period [Axis]		
	2013-01-01 - 2013-12-31	2012-01-01 - 2012-12-31	2011-01-01 - 2011-12-31
Balance at beginning of period	26,987,000	12,742,000	8,972,000
Acquisition			3,151,000
Deferral of new extended warranty revenue	20,191,000	22,344,000	8,659,000
Recognition of extended warranty deferred revenue	(12,789,000)	(8,099,000)	(8,040,000)
Balance at end of period	34,389,000	26,987,000	12,742,000

All the facts in this set forms a Block and that Block, as we have said, is made up of one concept arrangement pattern. In this case the pattern is that of a roll forward, rather than that of a roll up which was shown in the first example.

A roll forward does not have a total. It may look like a total to the untrained eye, but rather than totaling a set of facts, a roll forward reconciles a fact between two periods for the changes between the two periods. The concept used to describe the fact used to represent the beginning and ending balance is the same for a roll forward; but the period of the concept is different for the beginning and ending balances. It would make no sense to have a third period for one roll forward. It would likewise make no sense to have the concept "Land" as a participant of the changes that are being represented by this roll forward.

Further, consider the third example below. The fragment of a financial report shown below consists of two Blocks. The first block "Restructuring Reserve [Roll Forward]" is a roll forward and the second Block "Restructuring Reserve [Roll Up]" is a roll up.

As was said, Blocks share the same concept arrangement pattern. The concept arrangement pattern is shown in the line items or rows on the left. Notice though

that both Blocks share the same member arrangement pattern. The [Member]s of an [Axis] form patterns also. What you see below is the "Restructuring Type [Axis]" being used to detail the [Member]s "Facility Closing [Member]", "Severance [Member]", and the total "All Restructuring Types [Domain]". That member arrangement pattern is applicable to both the roll forward and roll up which are provided.

	Period [Axis]					
	2010-01-01 - 2010-12-31			2009-01-01 - 2009-12-31		
	Restructuring Type [Axis]			Restructuring Type [Axis]		
Restructuring Cost and Reserve [Line Items]	Facility Closing [Member]	Severance [Member]	All Restructuring Types [Domain]	Facility Closing [Member]	Severance [Member]	All Restructuring Types [Domain]
Restructuring Reserve [Roll Forward]						
Restructuring reserve, beginning balance	97,000,000	204,000,000	301,000,000	94,000,000	200,000,000	294,000,000
Restructuring charge	(1,000,000)	0	(1,000,000)	(4,000,000)	(4,000,000)	(8,000,000)
Cash payments	(4,000,000)	(4,000,000)	(8,000,000)	(6,000,000)	(6,000,000)	(12,000,000)
Accrual adjustment	0	(1,000,000)	(1,000,000)	(1,000,000)	0	(1,000,000)
Translation adjustment	30,000,000	5,000,000	35,000,000	14,000,000	14,000,000	28,000,000
Restructuring reserve, ending balance	122,000,000	204,000,000	326,000,000	97,000,000	204,000,000	301,000,000
Restructuring Reserve [Roll Up]						
Current portion of restructuring reserve	96,000,000	204,000,000	300,000,000	96,000,000	204,000,000	300,000,000
Long-term portion of restructuring reserve	26,000,000	0	26,000,000	1,000,000	0	1,000,000
Restructuring reserve	122,000,000	204,000,000	326,000,000	97,000,000	204,000,000	301,000,000

In order to maintain a one-to-one correlation between a piece of the report and the concept arrangement pattern used to represent the piece of the report, the notion of the *block* is used. By doing so you can talk about or work with the roll forward at the top or the roll up at the bottom as unique objects.

By thinking of the one report fragment as two blocks, each with a unique identifiable concept arrangement pattern and specific member arrangement pattern; software can be created which helps business professionals using and creating the information contained within a digital financial report.

Accountants have the option of combining and presenting information in different ways when they want to providing disclosures. But accountants have far fewer options when it comes to representing the information in logical, sensible, and mathematically and mechanically correct ways.

Not understanding the information makes it harder to create and harder to use the information because accountants have to work with lower-level pieces which are more technical in nature and therefore harder for business professionals to work with.

We will provide one final example in this section to help solidify the notion of a Block. Below you see four blocks: the first two are [Roll Forward]s, the third a [Roll Up], and the fourth a [Hierarchy]. The two [Roll Forward]s are connected to the [Roll Up], the ending balances of the [Roll Forward]s are the items which are being rolled up in the [Roll Up]. Because the information is represented correctly and because the rendering engine which produced the renderings from the machine-readable representation understands the notion of a Block, the information is easy for both a machine reading the information or for a human reading the information to understand.

In addition to the concept arrangement patterns which show the organization of the [Line Items] (which are in the rows on the left of the rendering), the information is further distinguished using the *Defined Benefit Plan Category [Axis]*.

A block is a combination of a *concept arrangement pattern* and *member arrangement patterns* which work together to distinguish reported facts.

	Period [Axis]					
	2011-01-01 - 2011-12-31			2010-01-01 - 2010-12-31		
	Defined Benefit Plan Category [Axis]			Defined Benefit Plan Category [Axis]		
Defined Benefit Plan Disclosure [Line Items]	U.S. Pension Benefits [Member]	Non-U.S. Pension Benefits [Member]	Other Postretirement Benefits [Member]	U.S. Pension Benefits [Member]	Non-U.S. Pension Benefits [Member]	Other Postretirement Benefits [Member]
Change in benefit obligation [Roll Forward]						
Benefit obligation at beginning of year	444,000,000	593,000,000	166,000,000	375,000,000	327,000,000	157,000,000
Service cost	38,000,000	9,000,000	8,000,000	32,000,000	8,000,000	10,000,000
Interest cost	21,000,000	33,000,000	8,000,000	22,000,000	26,000,000	9,000,000
Actuarial loss	43,000,000	25,000,000	28,000,000	31,000,000	4,000,000	10,000,000
Benefits paid	(19,000,000)	(16,000,000)	(14,000,000)	(47,000,000)	(12,000,000)	(15,000,000)
Curtailment	0	(4,000,000)	0	0	(1,000,000)	0
Acquisitions of businesses	0	2,000,000	0	34,000,000	253,000,000	27,000,000
Plan amendments	0	0	0	0	0	(32,000,000)
Other changes	(3,000,000)	1,000,000	0	(3,000,000)	2,000,000	0
Exchange rate adjustments	0	0	0	0	(14,000,000)	0
Benefit obligation at end of year	524,000,000	643,000,000	196,000,000	444,000,000	593,000,000	166,000,000
Change in plan assets [Roll Forward]						
Fair value of plan assets at beginning of year	416,000,000	474,000,000	0	346,000,000	248,000,000	0
Actual return on plan assets	(5,000,000)	38,000,000	0	48,000,000	36,000,000	0
Employer contributions	43,000,000	28,000,000	14,000,000	72,000,000	52,000,000	15,000,000
Acquisitions of businesses	0	0	0	0	160,000,000	0
Administration expenses	(2,000,000)	1,000,000	0	(3,000,000)	1,000,000	0
Exchange rate adjustments	0	1,000,000	0	0	(11,000,000)	0
Fair value of plan assets at end of year	433,000,000	526,000,000	0	416,000,000	474,000,000	0
Funding Status [Roll Up]						
Fair value of plan assets	433,000,000	526,000,000	0	416,000,000	474,000,000	0
Benefit obligation	524,000,000	643,000,000	196,000,000	444,000,000	593,000,000	166,000,000
Funded status - underfunded at end of year	(91,000,000)	(117,000,000)	(196,000,000)	(28,000,000)	(119,000,000)	(166,000,000)
Accumulated Benefit Obligation [Hierarchy]						
Accumulated benefit obligation	491,000,000	616,000,000	196,000,000	421,000,000	553,000,000	166,000,000

Your understanding of Blocks will increase as we fill in more and more details of the conceptual model of a digital financial report.

1.5. Understanding the notion of intersections

A financial report contains facts, those facts are organized within components which are than organized into blocks, and facts can exist within multiple components/blocks. For example, consider this balance sheet fragment and the related disclosure of property, plant, and equipment which shows one fact shown in two components/blocks:

Reporting Entity [Axis]	0000000001 (http://www.sec.gov/CIK)	
Legal Entity [Axis]	Consolidated Entity [Domain]	
	Period [Axis]	
Balance Sheet [Line Items]	2012-12-31	2011-12-31
Assets [Roll Up]		
Current assets [Roll Up]		
Cash, cash equivalents, and marketable securities [Roll Up]		
Cash and cash equivalents	11,000,000	10,000,000
Marketable securities	9,000,000	10,000,000
Cash, cash equivalents, and marketable securities	20,000,000	20,000,000
Accounts receivable, net of allowance for doubtful accounts of $1,000 and $1,000	29,000,000	29,000,000
Inventories	4,000,000	4,000,000
Prepaid expenses	3,000,000	3,000,000
Total current assets	56,000,000	56,000,000
Noncurrent assets [Roll Up]		
Property, plant and equipment, net	82,000,000	82,000,000
Deferred costs	9,000,000	9,000,000
Total noncurrent assets	91,000,000	91,000,000
Total assets	147,000,000	147,000,000
Liabilities and Equity [Roll Up]		
Current liabilities [Roll Up]		
Accounts payable	3,000,000	3,000,000
Accrued liabilities	4,000,000	4,000,000
Current portion of long-term debt	22,000,000	22,000,000
Product warranty accrual, current portion	26,000,000	26,000,000

Contrast the balance sheet to the disclosure below which shows the property, plant, and equipment breakdown:

Reporting Entity [Axis]		0000000001 (http://www.sec.gov/CIK)	
Legal Entity [Axis]		Consolidated Entity [Domain]	
		Period [Axis]	
Property, Plant and Equipment [Line Items]	Property, Plant and Equipment, Type [Axis]	2012-12-31	2011-12-31
Property, Plant and Equipment, Net, by Type [Roll Up]			
Property, plant and equipment, gross	Land [Member]	40,000,000	40,000,000
	Machinery and equipment [Member]	50,000,000	50,000,000
	Furniture and fixtures [Member]	7,000,000	7,000,000
	Property, Plant and Equipment, All Types [Domain]	97,000,000	97,000,000
Accumulated depreciation	Property, Plant and Equipment, All Types [Domain]	(15,000,000)	(15,000,000)
Property, plant, and equipment, net	Property, Plant and Equipment, All	82,000,000	82,000,000

The fact is the same. In both cases the fact is "property, plant, and equipment, net" which has the value 82,000,000. However the characteristics which describe the fact are different. On the balance sheet, there is no property, plant and equipment type [Axis], but in the disclosure there is. This is because the component which represents the disclosure needs to be able to differentiate the concepts.

It is challenging to show[1] the notion of an intersection and how useful it is in software applications without using software. This is best experienced to be fully appreciated.

> HINT: A good way to view intersections is using the free XBRL Cloud Viewer.

An intersection is where a fact in one component/block also exists in another component/block creating in essence a link between the two components, an intersection.

> HINT: Note that the actual fact exists only once physically, but XBRL Dimensions syntax rules work in such a manner that the fact exists virtually in two or more components/blocks.

1.6. Realizing that creating a financial report is about organizing blocks and adding things to blocks using slots

The structural pieces which make up a digital financial report can be grouped into high-level types or classes. No new high-level types or classes can be added, you may only use existing high-level types or classes. These high-level types or classes may never be redefined; you cannot arbitrarily change the meaning of a type or class. However, sub-types or subclasses can be created and added and identified as being associated with one of those existing high-level classes of things. But sub-types/subclasses can only be added as specified by the system. Individuals can be created and specified as being a member of one type/class or another, you simply cannot create an individual which is associated with nothing or which is two things at the same time.

And so:

- **Adding new economic/accounting entities**: An economic/accounting or reporting entity is created by creating a new instance of identifier. For example the CIK number of a public company which reports to the SEC.
- **Adding new report**: A new report is created by creating a new report instance. For example, Microsoft submits a new financial report in order to report information for fiscal year ended 2014.
- **Adding a new characteristic:** A new characteristic can be added but the characteristic MUST be distinguished as being either a "whole-part" or "is-a" type of relation or some existing subclass of existing relations (which must be one of those two). For example, Microsoft uses the existing characteristic "Legal Entity [Axis]" or Microsoft creates the characteristic "Tax Entity [Axis]" and distinguishes that characteristic as being a "whole-part" type of relation.
- **Adding new concept characteristic:** A new concept can be added but the concept MUST be distinguished as being a subclass of some existing concept or distinguished as being a new type of class (if that is allowed). For example, Microsoft might add a new concept to its balance sheet such as "Ultra-tangible asset"; however it MUST NOT break the rules of a "roll up" because a balance sheet is a roll up. Further, the added concept MUST be identified as a subclass of something that exists on a balance sheet which can contain ONLY assets, liabilities, or equity.

[1] This video walks you through the notion of an intersection, https://www.youtube.com/watch?v=lNPjwKy2Obs

- **Adding new disclosure (component or block):** A disclosure is in essence a set of facts which must be disclosed. A set of facts is represented as a component and that component might have one or many blocks. To add a new disclosure, a reporting entity simply creates a new component and/or block individual. That individual of the class component MUST be (i.e. follow) the relations patterns of the existing component which the individual is a member of. For example, if Microsoft creates a "balance sheet" individual, it must associate that individual with the existing class "balance sheet" and therefore must follow the relation rules of a roll up because the existing component "balance sheet" is a roll up. Why? Because a balance sheet is a roll up, it is not ever a roll forward. Now, a reporting entity could also, if they desired, create a new subclass of "balance sheet" called "my balance sheet" and associate it with the class "balance sheet". Or, a company could create an entirely new disclosure such as "cash and cash equivalents by county", associate that disclosure not with some existing disclosure but rather with the root class "component" and then provide a completely new disclosure. However, what the reporting entity may NOT do is create some new relations pattern, it must use existing relations patterns (i.e. no new relation patterns can be added). Basically, any individual MUST follow the rules as must any new class.
- **Adding facts**: A fact is always an individual. Facts are put into blocks which go into components. Facts are never "free floating in space". Every fact has distinguishing aspects to make them identifiable from other reported facts. Facts are described by characteristics, exist within a report, and are reported by an economic/accounting entity. For example, the accounting entity Microsoft might report the fact 1,000,000 which relates to the consolidated entity, to the current balance sheet date of December 31, 2014, be reported in US Dollars, and report the balance sheet line item Cash and cash equivalents. That fact might be in the component balance sheet and has a relation between the concept Current assets in that it rolls up to that total.
- **Adding new properties**: New properties MUST NEVER be added, XBRL-based financial filings to the SEC does not allow the addition of new properties, there is no "slot" available where new properties may be added. For example, a report element does have the property "balance type" (an XML Schema attribute); but a new attribute such as "my attribute" cannot be created and added.

Different systems can have different rules for allowing new classes, sub-types/subclasses, relations between classes, or properties. System boundaries can be extended by adding new relation patterns. New relation patterns must be consciously and formally added in a controlled and coordinated manner only by system implementers before any new pattern is allowed to be used. System boundaries can be extended by adding new classes or properties. New classes and new properties must be consciously and formally added in a controlled and coordinated manner only.

1.7. Understanding why adding new patterns is both rare and not a constraint

Adding new patterns is both rare and not a constraint. While this notion might seem absurd or unintuitive, it is important to look at empirical evidence to understand why

this is the case. Ultimately, a pattern can be identified for anything. The real question is this: How wide-spread is the use of the pattern.

But this is not to say that a new pattern cannot be added. The point here is to say that adding patterns is rare and it is a different process from using existing patterns.

If one were to observe XBRL-based financial filings, one would realize that 100% of public company financial reports contain [Line Items] which contain concepts and abstracts which follow these *concept arrangement patterns*:

- **Text block**
  - Level 1 Note Level Text Block
  - Level 2 Policy Level Text Block
  - Level 3 Disclosure Level Text Block
- **Roll Up**: Concept A + Concept B + Concept *N* = Total
- **Roll Forward**: Beginning balance + Additions – Subtractions – Ending balance
- **Hierarchy**: No mathematical relationships
- **Adjustment**: Originally stated balance + Adjustments = Restated balance
- **Roll Forward Info**: Beginning balance info + Additions info – Subtractions info – Ending balance info (there are no mathematical relations, but information for the beginning and ending balances must be distinguished)

Similarly, each [Axis] falls into one of two broad categories and describes the [Member]s of that [Axis] as being one of the following two *member arrangement patterns*:

- **Whole-part**: Characteristic describes something composed exactly of their parts and nothing else or more where the parts add up to the whole
- **Is-a**: Characteristic describes some list but the list does not add up mathematically

If some other concept arrangement pattern or member arrangement pattern is identified; then that pattern or patterns can be added to the set of patterns above.

Consider the following theory: A combination of those *concept arrangement patterns* and *member arrangement patterns* describes every component of every report of every reporting entity which submits XBRL-based financial information to the SEC.

Being conservative, that theory is already proven to be true for 98% of the components of public company financial reports. By being conservative, we leave open the possibility for 2% of report components which might deviate from these patterns above for because they are not structural patterns described in this document. Basically, the following spectrum delineates all possible alternatives:

1. A reporting entity report component follows (**is consistent with**) existing concept arrangement patterns and existing member arrangement patterns.
2. A reporting entity component DOES NOT FOLLOW, however SHOULD FOLLOW (**is inconsistent with**) existing concept arrangement patterns and existing member arrangement patterns. HOWEVER, after the inconsistency is

corrected within the report, the reporting entity report component follows (**is consistent with**) existing patterns.

3. A reporting entity component DOES NOT FOLLOW, but either a concept arrangement pattern or member arrangement pattern IS MISSING from the list of allowed patterns. The pattern is logical, rational and sensible; plus it does not cause some logical catastrophe. THEREFORE, the pattern should be added (**is consistent with**).
4. A reporting entity component DOES NOT FOLLOW, but either a concept arrangement pattern or member arrangement pattern IS MISSING from the list of allowed patterns. The pattern is logical, rational and sensible HOWEVER; the pattern (a) can be reduced down to a less complex pattern and (b) if added it WOULD cause some sort of logical catastrophe. THEREFORE, the pattern should NOT BE ADDED. Rather, the reporting entity should change how they represent information within a report to keep the overall system safe, reliable, predictable (**is consistent with**).
5. A reporting entity component follows (**is consistent with**) the existing [Hierarchy] concept arrangement patterns and an existing member arrangement pattern; HOWEVER the pattern is in reality not a [Hierarchy] but rather some other unsupported mathematical relation or some other unsupported member arrangement pattern. While not optimal because specific information which could be verified to be consistent is not being verified, this is still on par with current practices. Currently, a [Roll Forward] is a known and a commonly used pattern. The pattern is identifiable, but has no computation articulated within XBRL taxonomies which support XBRL instances submitted to the SEC.

And so either a filer is already consistent with the existing system (#1), or should be consistent with the existing system (#2). It is possible that a reporting entity is using a logical and sensible concept arrangement pattern or member arrangement pattern that is missing (#3); and if so, that pattern should be added to the system. It is possible that a reporting entity is using a logical and sensible concept arrangement pattern or member arrangement pattern; however, (a) that pattern can be broken down into a simpler, less complex pattern and (b) if the pattern were added to the system it would make the system unsafe, unpredictable, or otherwise unreliable and therefore should not be added to the system.

And, as discussed in the next section, there is always a fallback position (#5). Everything can be represented as a [Hierarchy] concept arrangement pattern. Other concept arrangement patterns simply add additional rules, generally mathematical computations. This allows new patterns to evolve. This is explained in more detail in the next section.

1.8. Understanding that pattern maintenance is an evolutionary process

Every concept arrangement pattern is some [Hierarchy][2] of concepts. Other non-[Hierarchy] concept arrangement patterns add some sort of mathematical computation. For example,

[2] I really don't like the name [Hierarchy] because everything is a hierarchy. A better term might be [Set] or some other term.

- A [Roll Up] is simply a [Hierarchy] with the addition of XBRL calculation relations which articulate the information about how the concepts roll up.
- A [Roll Forward] is simply a [Hierarchy] with the addition of a preferred label role to differentiate the beginning and ending instant concept.
- An [Adjustment] is simply a [Hierarchy] with the addition of a preferred label role to differentiate the originally stated and restated balances plus a member arrangement pattern to distinguish the Report Date [Axis].
- A [Text Block] is a [Hierarchy] which has only one concept which is of a specific data type.

Basically, any information can be represented as a [Hierarchy]. The down side of representing information in this manner if it really is some other pattern is that you do not provide necessary rules which software can use to assure that what is represented is consistent with reality. The information might be consistent with the knowledgebase of information, but that is only because the rules are not included in the knowledgebase. What that means is that the information needs to be verified using manual processes because consistency cannot be determined using automated processes because there are no machine-readable business rules.

This situation is not optimal, but it is also not the end of the world either. As was stated above, this situation is on par with current XBRL-based public company financial filings in that [Roll Forward] concept arrangement patterns in existing SEC filings do not provide business rules for the [Roll Forward].

What this means is that there is already a process to allow patterns to evolve.

1.9. Understanding that patterns are finite (i.e. not infinite)

To understand that it is not an overwhelming task to inventory all patterns and add new patterns to the system, consider the notion of report style patterns[3]. If you look closely at the reporting style patterns, this is what you observe:

- Every public company can be grouped into one of approximately 100 reporting styles.
- Of the approximately 7,000 reporting entities in scope (funds and trusts are excluded as they follow other patterns which are not of interest); 90% of all public companies fall into one of 15 reporting styles. The remaining 10% of reporting entities use the other 85 reporting styles.
- Some of the reporting styles which are used are likely illegal. For example, why would a commercial and industrial company report using an unclassified balance sheet? Meaning, some existing reporting styles need to be removed.
- It is highly-likely that some reporting styles will have only 1 reporting entity which makes use of that reporting style. For example JPMorgan seems to fall into that category. Nothing wrong with that.
- It is highly-likely that there are between perhaps 100 to maybe even 250 additional reporting styles. It is of no consequence to have 100 or even 250 or maybe even 500 additional reporting styles.

[3] For more information on report frame patterns, see http://www.xbrlsite.com/2015/fro/us-gaap/html/ReportFrames/

Every other type or class works precisely the same way. Some finite list of sub-types/subclasses can exist. And so, the system is finite, the system has boundaries, but the system is flexible but only where specific flexibility is exposed.

1.10. Understanding technical syntax rules and workflow/process rules

There has not really been much emphasis on technical syntax rules and workflow/process rules, the primary focus is on business domain rules.

The reason for less effort in explaining technical syntax rules is because of the following:

- XBRL technical syntax rules were created and interoperability between software is excellent due to a publically available conformance suite provided by XBRL International.
- Because of the first point; XBRL-based digital financial reports provided to the SEC by public companies are 99.99% consistent with the XBRL technical syntax rules.
- Business professionals should never be exposed to technical syntax; software should hide all aspects of technical syntax from business professionals.

Basic workflow/process rules are worth covering a little because that would yield important useful information. However, there has not been a lot of focus on workflow/process rules so we really don't know the full extent of what workflow/process rules are necessary.

However, we do understand the basic, fundamental rules which are necessary for any system to work with a digital financial report.

Consider a simple query of two concepts: *Assets* and *Liabilities and Equity*. In order to extract that information from any XBRL-based financial report using a machine-based process the following process needs to be followed:

1. Software MUST locate each report you want to query.
2. The report MUST be valid XBRL technical syntax. If the technical syntax is invalid, you may or may not get the correct results.
3. Software MUST locate the appropriate reporting units (currency). In the case of public company financial reports, 99% of entities report using US Dollars. However, 1% use other currencies as the reporting units.
4. Software MUST appropriately identify the root reporting entity in the report. Generally, this is the consolidated entity but it could be a parent holding company or some other accounting entity.
5. Software MUST appropriately locate the current balance sheet date. Generally you want information about the current balance sheet information and not the prior balance sheet.
6. Software MUST find the appropriate fact which uses the US GAAP concept *Assets* which is us-gaap:Assets.
7. Software MUST find the appropriate fact which uses the US GAAP concept for *Liabilities and Equity*. This is a little harder because there are multiple

possible concepts: us-gaap:LiabilitiesAndStockholdersEquity or us-gaap:LiabilitiesAndPartnersCapital.

8. Software MUST check the returned information to assure that it is consistent with what is expected, the business domain rule that "Assets = Liabilities and Equity".

That is an overview of the workflow/process to obtain a basic set of information from the knowledgebase of XBRL-based public company financial filings. And here are the results of that query for every financial report in that data set:

xbrl:Entity	Legal Entity	Fiscal Period	Fiscal Year	Assets	Liabilities and Equity	Units	Difference in Value
All CIK numbers	Root economic entity	FY	2001	280	280	iso4217:USD	0
All CIK numbers	Root economic entity	FY	2009	31,586,555,000	31,586,555,000	iso4217:USD	0
All CIK numbers	Root economic entity	FY	2010	23,061,516,000	23,061,516,000	iso4217:CAD	0
All CIK numbers	Root economic entity	FY	2010	8,833,200,000	8,833,200,000	iso4217:GBP	0
All CIK numbers	Root economic entity	FY	2010	33,205,444,569,755	33,235,543,477,631	iso4217:USD	30,098,907,876
All CIK numbers	Root economic entity	FY	2011	45,216,467	45,216,467	iso4217:AUD	0
All CIK numbers	Root economic entity	FY	2011	110,885,000	110,885,000	iso4217:BRL	0
All CIK numbers	Root economic entity	FY	2011	28,708,716,218	28,708,716,218	iso4217:CAD	0
All CIK numbers	Root economic entity	FY	2011	1,226,733,000	1,226,733,000	iso4217:EUR	0
All CIK numbers	Root economic entity	FY	2011	7,938,800,000	7,938,800,000	iso4217:GBP	0
All CIK numbers	Root economic entity	FY	2011	1,565,000	1,565,000	iso4217:ILS	0
All CIK numbers	Root economic entity	FY	2011	46,395,324,314,234	46,165,763,878,111	iso4217:USD	(229,560,436,123)
All CIK numbers	Root economic entity	FY	2012	49,066,850	49,066,850	iso4217:AUD	0
All CIK numbers	Root economic entity	FY	2012	32,470,161,238	32,470,161,238	iso4217:CAD	0
All CIK numbers	Root economic entity	FY	2012	1,303,349,000	1,303,349,000	iso4217:EUR	0
All CIK numbers	Root economic entity	FY	2012	10,504,300,000	10,504,300,000	iso4217:GBP	0
All CIK numbers	Root economic entity	FY	2012	47,493,211,088,244	47,307,285,874,940	iso4217:USD	(185,925,213,304)
All CIK numbers	Root economic entity	FY	2013	54,642,443	54,642,443	iso4217:AUD	0
All CIK numbers	Root economic entity	FY	2013	39,919,462,935	39,919,385,738	iso4217:CAD	(77,197)
All CIK numbers	Root economic entity	FY	2013	13,120,000	13,120,000	iso4217:EUR	0
All CIK numbers	Root economic entity	FY	2013	48,909,115,040,682	48,735,740,980,605	iso4217:USD	(173,374,060,077)
All CIK numbers	Root economic entity	FY	2014	342,493,649,881	342,493,649,881	iso4217:USD	0
				176,531,415,952,227	175,972,655,073,402		(558,760,878,825)
							-0.3%

The results[4] show that most of the balance sheets balance, Assets = Liabilities and Equity. Some are inconsistent with what you would expect. The total inconsistency is .3% which is not too bad. However, the information needs to be much closer to 100% consistent in order to not get humans involved to figure out what is causing the inconsistencies.

1.11. Proving the structural mechanics using XBRL-based public company financial filings

How can you tell if the mechanics that this paper describes is correct? It is actually rather easy: look at publically available XBRL-based financial filings which public companies report to the SEC. First though, you need to reconcile the mechanical representation with an implementation of the mechanical representation in software[5]. While it is beyond the scope to do a detailed reconciliation between the semantics use in this document, the terms use by software and the US GAAP XBRL

[4] Query and results provided by SECXBRL.info which is a commercial software application, see http://app.secxbrl.info/

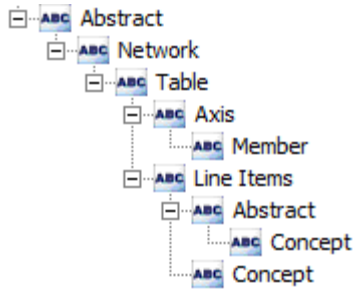
[5] Reconciliation of Financial Report Semantics and Dynamics Theory, to US GAAP XBRL Taxonomy Architecture and SEC implementation, to XBRL technical syntax, see http://www.xbrlsite.com/2012/Library/SemanticObjectlReconciliation.pdf

Taxonomy and SEC, and the XBRL technical syntax specification; it is necessary to provide an overview because we need to shift terminology slightly. This is that overview which reconciles terminology:

Term used in this document	**Term used by software**
Economic or accounting entity	Reporting Entity CIK (XBRL context entity identifier)
Report	XBRL instance document + XBRL taxonomy
Component	XBRL **Network** + [**Table**]
Characteristic (other than concept)	[**Axis**] + [**Member**]
Characteristic (concept)	[**Line Items**] + **Concept**
Fact	Fact
Block	XBRL Network + [**Table**] + [**Abstract**]
Relations pattern	NOT IN SCOPE
Properties	NOT IN SCOPE

That is a rough explanation of the terms we use to describe the mechanics of a financial report and terms use by software applications, XBRL-based financial filings of public companies to the SEC, and the US GAAP XBRL Taxonomy. A complete reconciliation of terminology is beyond the scope of this document and would cause more confusion and complexity that most business professionals would tolerate.

To keep this simple, the implementation of the mechanics can be distilled down to the following classes of report elements: Network, Table, Axis, Member, Line Items, Abstract and Concept. They are roughly related as follows:



Software was used to query the mechanical structure of 6,674 XBRL-based public company 10-K filings for primarily fiscal year 2013 and the following results were obtained:

		Parent						
		Network 477,041	Table 232,230	Axis 386,912	Member 1,216,391	LineItems 232,690	Abstract 732,409	Concept 3,165,249
Child	Network	0	0	0	0	0	0	0
	Table	1,261	1	0	0	45	230,899	24
	Axis	1	386,888	0	0	3	20	0
	Member	3	0	450,091	766,221	4	72	0
	LineItems	183	232,181	0	0	107	217	2
	Abstract	474,310	22	0	1	113,059	144,471	546
	Concept	46	26	11	137	1,222,427	1,929,257	13,346

In the columns are the structural type/class of pieces which serve as the parent for some child structural type/class of piece: Network, Table, Axis, and so on. In the rows are the child structural pieces: Network, Table, Axis, and so on. The cells show the number of relations which exist in the set of 6,674 digital financial reports.

This second graphic of the same information will help you to interpret and understand the results:

		Parent						
		Network 477,041	Table 232,230	Axis 386,912	Member 1,216,391	LineItems 232,690	Abstract 732,409	Concept 3,165,249
Child	Network	0	0	0	0	0	0	0
	Table	1,261	1	0	0	45	230,899	24
	Axis	1	386,888	0	0	3	20	0
	Member	3	0	450,091	766,221	4	72	0
	LineItems	183	232,181	0	0	107	217	2
	Abstract	474,310	22	0	1	113,059	144,471	546
	Concept	46	26	11	137	1,222,427	1,929,257	13,346

What the graphic says about the relationships between the structural pieces of the digital financial reports is the following:

- Of the 386,912 [Axis] which exist in the report, there are ZERO occasions where a parent [Axis] has a child [Axis].
- Of the 232,690 [Line Items] which exist in the report, there are 1,222,427 occasions where the parent [Line Items] has a child which is a Concept.

Without going into a lot of detail, the following graphic shows what the above graphic means: the allowed and disallowed relations between the mechanical building blocks: Network, Table, Axis, Member, LineItems, Abstract and Concept.

		Parent						
		Network	Table	Axis	Member	LineItems	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
	LineItems	Disallowed	OK	Disallowed	Disallowed	Disallowed	Disallowed	Disallowed
	Abstract	OK	Disallowed	Disallowed	Disallowed	OK	OK	Not advised
	Concept	Not advised	Disallowed	Disallowed	Disallowed	OK	OK	Not advised

The point here is not to have a debate about what should be allowed and what should not be allowed. While that debate and perhaps even a theoretical or philosophical discussion about the merits of allowing or disallowing relations could prove useful, that is not the point.

The point is this: First, if a profound majority of XBRL-based financial reports are represented in a certain way, it is very difficult to say that the approach is wrong. Not impossible because the majority could be incorrect in certain occasions.

But second, and most importantly, if rules can be created and enforced by software and it is possible to have 100% agreement then why is that not done?

Look at the graphic again. Notice that there are ZERO occasions where a Network is a child of any other mechanical structure. Why is that? The reason that there are ZERO is that the XBRL technical specification states that such relations are not allowed, and the XBRL consistency suite tests to make sure software does not make this mechanical mistake. And software vendors test their software against the XBRL consistency suite. Therefore, no software has these sorts of errors and is therefore interoperable with other software applications.

And so an obvious question is this: why are not other structural aspects not enforced in this manner?

1.12. Proving other mechanics using XBRL-based public company financial filings

The following is a summary of the consistency of other logical, mechanical, or mathematical rules of XBRL-based public company 10-K financial filings from the same set of 6,674 filings for FY 2013, an earlier set of similar 10-K financial reports for FY 2012, and for another similar set of 10-Q and 10-K financial filings for FY 2014[6].

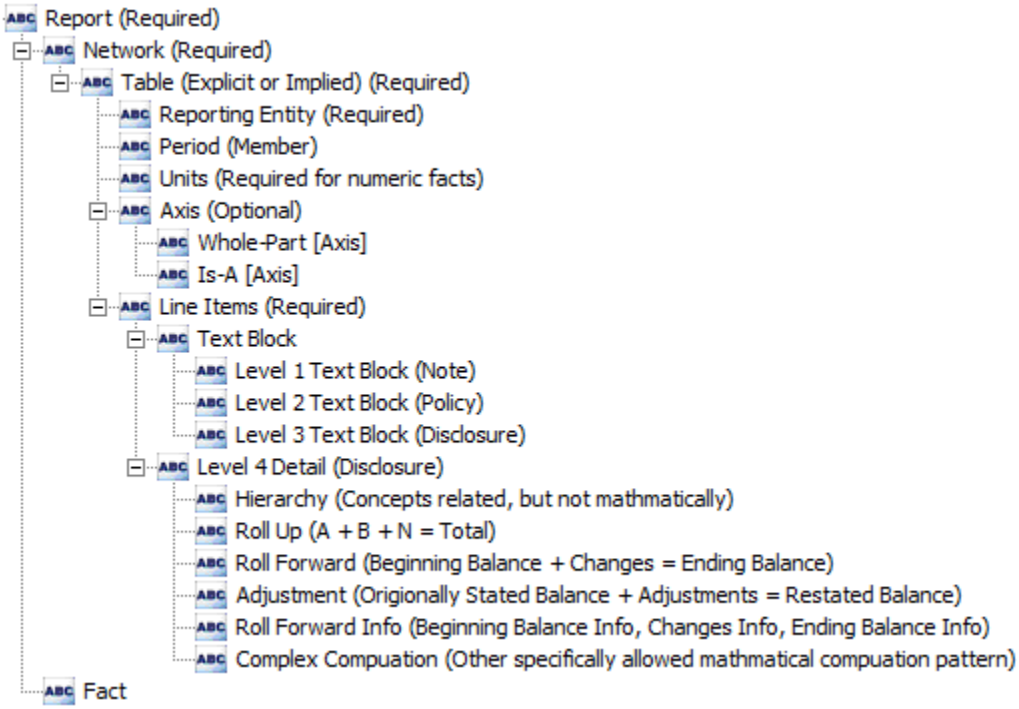
#	Goal or Desired State	Process tests	FY 2014	FY 2013	FY 2012
1	Consistent XBRL technical syntax	Automated XBRL technical syntax error checks	99.9%	99.9%	99.9%
2	Consistent EDGAR Filer Manual (EFM) syntax/semantics	Automated EFM syntax and semantics error checks	98.0%	97.9%	80.5%
3	Consistent report level structure	Automated model structure error checks	97.6%	95.8%	97.9%
4	Detectable economic entity or accounting entity or "root reporting entity" or "entity of focus"	Successful and unambiguous identification of the "entity of focus"	99.3%	99.2%	98.8%
5	Detectable and unambiguous current period balance sheet and income statement period dates	Successful and unambiguous identification of the current balance sheet date and income statement period	98.4%	99.3%	99.8%
6	Detectable and unambiguous set of fundamental reported facts and intact relations between those fundamental facts which prove trustworthy nature of information	Automated verification checks to be sure fundamental accounting concepts are distinguishable/decipherable and the relations between those fundamental concepts are intact/sound	97.9%	97.8%	97.9%
7	Detectable basic primary financial statement roll up computations are intact which prove trustworthy nature of information	Automated verification checks for existence of business rules which articulate these basic primary financial statement relations and successful passing of these business rules	90.7%	90.1%	84.9%

The primary point here is that if you look at the columns on the right for FY 2014, FY 2013, and FY 2012 you notice that testing against what we would expect yielded a very high number of XBRL-based public company financial reports that are consistent with those expectations.

[6] Not all FY 2014 financial filings have been submitted to the SEC as of the date of this document, so the latest 10-Q was used if the 10-K was not available.

1.13. Summary of the complete representation model and mechanics

To tie all of the pieces together, we provide this summary of the representation model and an overview of the logical, mechanical and mathematical aspects of a financial report. The graphic below shows each of the implementation pieces which can be different depending upon how a software application exposes the pieces of a digital financial report to its business users. This is a summary of the pieces of a financial report.



The table below summarizes the pieces that exist in the 10-K financial information of 6,674 public companies who report to the SEC using the XBRL format. The class of report piece, a count of the individuals in those reports, an average for many of the pieces and a brief comment is provided:

Class	Count	Average per Report	Comment
Report	6,674	1	Facts required to exist in Report
Network	477,041	71	Part of Component
Table	232,230	35	Part of Component
Axis	386,912	58	Part of Characteristic
Member	1,216,391	181	Part of Characteristic

Line Items	232,690	35	Type of [Axis], subclass of Characteristic
Abstract	732,409	111	No meaning, only used for organization
Concept	3,165,249	474	Part of Characteristic
Properties	Not counted		Each class has different but finite properties
Fact	8,532,275	1,278	Described by Characteristic, Required to exist within Network, Required to exist within explicit or implied Table
Text Block	398,492	59	Counted facts with data type of nonnum:textBlockItemType
Roll Forward	48,960		Counted preferred label roles which had start date and end date (approximate)
Roll Forward Info	18,794		Counted preferred label roles which had start date and end date but data type was not monetary (approximate)
Roll Up	114,584		Counted XBRL calculation relation roots
Hierarchy			Counted Networks with no matching XBRL calculation and no start date/end date preferred label role (work in progress)
Whole-part			Count specific [Axis] types (work in progress)
Is-a			Count specific [Axis] types (work in progress)

Taking this one step further, this provides lists of the next level of the digital financial report, the classes of text blocks, disclosures, characteristics, etc:

Class	**Comment**
Axis (need to break this out by whole-part and is-a type relations)	http://www.xbrlsite.com/2015/fro/us-gaap/html/Classes/Axes_Tree.html
Level 1 Note Level Text Blocks	http://www.xbrlsite.com/2015/fro/us-gaap/html/Classes/Level1TextBlock_Tree.html
Level 2 Policy Level Text Blocks	http://www.xbrlsite.com/2015/fro/us-gaap/html/Classes/Level2TextBlock_Tree.html

Level 3 Disclosure Level Text Blocks	http://www.xbrlsite.com/2015/fro/us-gaap/html/Classes/Level3TextBlock_Tree.html
Hierarchy	http://www.xbrlsite.com/LinkedData/Exemplars/Disclosures.aspx?InformationModel=[Hierarchy]
Roll Up	http://www.xbrlsite.com/LinkedData/Exemplars/Disclosures.aspx?InformationModel=[Roll Up]
Roll Forward	http://www.xbrlsite.com/LinkedData/Exemplars/Disclosures.aspx?InformationModel=[Roll Forward]
Report	http://www.sec.gov/Archives/edgar/monthly/xbrlrss-2014-12.xml

1.14. Summary of properties of high-level types/classes

The following is a summary of the relevant properties for each high-level type/class of report element:

- Report
  - Collection of networks
  - Collection of components
- Component
  - Collection of report elements
  - Collection of blocks
- Network
  - Name
  - Label (SortCode + Type + Title)
- Table
  - Name
  - Label
- Axis
  - Name
  - Label
- Member
  - Name
  - Label
- Line Items
  - Name
  - Label

- Abstract
  - Name
  - Label
- Concept
  - Name
  - Label
  - Data Type
  - Period Type
  - Balance Type
- Fact
  - Collection of characteristics
  - Fact value
  - Collection of parenthetical explanations
  - Units (numeric facts only)
  - Decimals (numeric facts only)

Note that while the XBRL technical syntax might require other properties (implemented as an attribute of an element), the properties are meaningless in terms of semantics. For example, a Table is required to have a data type of string, a period type of duration, and an abstract value of true; but that information is not relevant to the meaning of a Table.

While all report elements are required to have an ID attribute, that ID attribute is always identical to the namespace prefix plus "_" plus the name of the report element. Machines can automatically manage the ID.