

# Process Automation Using XBRL

## Elegant, safe, reliable process automation leveraging the global standard XBRL

By Charles Hoffman, CPA ([Charles.Hoffman@me.com](mailto:Charles.Hoffman@me.com))

Last Revised – May 16, 2018 (DRAFT)

“You never change things by fighting the existing reality. To change something, build a new model that makes the existing model obsolete.” R. Buckminster Fuller<sup>1</sup>

### Executive summary:

- Machine intelligence (sometimes called artificial intelligence) can be effectively leveraged to automate processes<sup>2</sup>. But what technical syntax would you use?
- XBRL is a well-thought-out flexible global standard syntax for representing facts and describing relations between facts. XBRL can handle financial and non-financial facts. XBRL excels at handling complex sets of facts.
- But XBRL can also be used for smaller sets of facts, binding information together in a tidy package, bound together with the strong “steel wires” of business logic.
- The basic example provided by this document helps you understand the possibilities.

---

<sup>1</sup> Good Reads, <https://www.goodreads.com/quotes/13119-you-never-change-things-by-fighting-the-existing-reality-to>

<sup>2</sup> Charles Hoffman, *Getting Ready for the Digital Age of Accounting, Reporting and Auditing: a Guide for Professional Accountants*, <http://xbrlsite.azurewebsites.net/2017/Library/GettingReadyForTheDigitalAgeOfAccounting.pdf>

### Copyright (full and complete release of copyright)

All content of this document is placed in the public domain. I 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 me. To be clear, I am granting full permission to use any content in this work in any way you like. I fully and completely release all my 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.

Fundamentally, XBRL is a well-thought-out flexible global standard syntax for representing facts and describing relations between facts. Many people believe that XBRL can only handle financial facts. However, XBRL can handle financial and non-financial facts. XBRL excels at handling complex sets of facts such as an entire financial report. But XBRL can also be used for smaller sets of facts, binding information together in a tidy package, bound together with the strong “steel wires” of business logic.

## ***General Application Profile of XBRL for Process Automation***

Think about what you would have to do if you wanted to implement business process automation. First you would have to establish a technical syntax for representing the business logic of the information. This is no trivial task. The semantic web stack<sup>3</sup> might be a viable choice to use as a basic syntax, but you still need to represent the underlying conceptual model and business semantics of your specific use case.

Another alternative is to simply use a general profile of XBRL. No implementation of XBRL ever uses 100% of the set of XBRL technical specification. Instead, those implementing XBRL create an application profile which specifies the allowed architecture of their individual implementation. What I did was take the best ideas of the best financial reporting implementation architectures<sup>4</sup>, steal those ideas and create what I refer to as the *general application profile of XBRL*.

The advantage of this is that you get a thoroughly tested high-quality application profile of XBRL that can handle extremely complex financial and non-financial information. This application has a general business reporting conceptual model that is separate from the financial report conceptual model.

This document explains this technique for using this general application profile of XBRL to implement business process automation<sup>5</sup>. Routine, highly repetitive, high-volume, error prone, mundane, and other such processes are the low hanging fruit which is ripe for the picking.

## ***Information Focus, Not Presentation Focus***

While many implementing XBRL-based systems for financial reporting tend to take paper processing systems and duplicate those and try and add on XBRL to the process, paper processing is not the best way to look at these sorts of systems. When an uninformed person looks at, say, an XBRL-based financial report which looks like a mess because the representation of the information is flawed one can get the impression that XBRL does not work very well. But should one look more deeply and sort out the representation problems associated with an improper mindset; then one can see that if XBRL-based systems are implemented appropriately they actually work very well<sup>6</sup>.

---

<sup>3</sup> Comparing and Contrasting the Semantic Web Stack and the XBRL Stack,  
[http://xbrl.azurewebsites.net/2017/Library/SemanticWebStack\\_XBRLStack.pdf](http://xbrl.azurewebsites.net/2017/Library/SemanticWebStack_XBRLStack.pdf)

<sup>4</sup> XBRL-based Digital Financial Reporting Profiles and General Business Reporting Profile,  
<http://xbrl.azurewebsites.net/2018/Library/Profiles-2018-01-24.pdf>

<sup>5</sup> Shivon Zilis and James Cham, Harvard Business Review, *The Competitive Landscape for Machine Intelligence*,  
<http://xbrl.squarespace.com/journal/2018/1/31/harvard-business-review-the-state-of-machine-intelligence.html>

<sup>6</sup> Disclosure Best Practices, <http://xbrl-site-app.azurewebsites.net/DisclosureBestPractices/DisclosureBestPractices.aspx?DisclosureName=BalanceSheet>

## ***Language of Graphs, Networks, and Trees***

An XBRL-based business report is a **directed acyclic graph** (DAG)<sup>7</sup>. Ghislain Fourny does an excellent job of describing these graphs in technical terms in his book *The XBRL Book: Simple, Technical, Precise*<sup>8</sup>.

**Graph theory**<sup>9</sup> is a useful communications tool. A graph is an abstract idea that does not really exist in the real world. But, graph theory can be used to describe and explain real world information and structural relationships that can be reduced to a graph which is a set of objects and the relations between the objects. The graph is not the real world; the graph is just used to explain how the real world works.

**Network theory**<sup>10</sup> is part of graph theory and is the study of graphs. A network can be defined as a graph in which objects and relations between objects have been given attributes such as a “name” or a “role”. A graph is defined in mathematical terms. Tools of **network science**<sup>11</sup> can be used to study these graphs.

A **tree**<sup>12</sup> is a special type of graph. A tree is what is called an undirected graph because the items in a tree are connected by exactly one path. This is important to understand because it means that trees are safer than other types of graphs which can contain cycles. For example, one type of cycle is an infinite loop.

A **forest** is a disjointed union of trees. Trees have **branches**. Trees have **leaves**.

This language of graphs and trees is important to understand. Software developers tend to understand and use this language. It is also important to understand that these rules by which graphs, networks, and trees operate are guided by the rules of mathematics, logic, and set theory<sup>13</sup>. A complete discussion of graphs, networks, trees, etc. is beyond the scope of this document but I will provide examples where an understanding is important or helpful.

## ***Basic Business Report***

In order to explain business reports, I created a very basic but robust business report using Lorem Ipsum<sup>14</sup>. Lorem Ipsum is simply dummy text that is used by the printing and typesetting industry.

I created this example using Lorem Ipsum for two specific reasons. First, most examples of XBRL relate to financial reporting and I wanted to show that while XBRL can be used for financial reporting; it can also be used for representing non-financial information. Second, many people working with XBRL taxonomies have preconceived notions of how a taxonomy should be created that is biased by the terms that are being represented. So in this report the terms themselves are basically nonsense random Lorem Ipsum auto-generated text. As such, it is impossible to project bias as to the correct or incorrect creation of the underlying taxonomy. As such, the logic that I define in the taxonomy is the only logic that exists to understand the information represented within the XBRL-based report.

---

<sup>7</sup> Wikipedia, *Directed Acyclic Graph*, [https://en.wikipedia.org/wiki/Directed\\_acyclic\\_graph](https://en.wikipedia.org/wiki/Directed_acyclic_graph)

<sup>8</sup> Ghislain Fourny, *The XBRL Book: Simple, Technical, Precise*, <http://ghislainfourny.github.io/the-xbrl-book/>

<sup>9</sup> Wikipedia, *Graph Theory*, [https://en.wikipedia.org/wiki/Graph\\_theory](https://en.wikipedia.org/wiki/Graph_theory)

<sup>10</sup> Wikipedia, *Network Theory*, [https://en.wikipedia.org/wiki/Network\\_theory](https://en.wikipedia.org/wiki/Network_theory)

<sup>11</sup> Wikipedia, *Network Science*, [https://en.wikipedia.org/wiki/Network\\_science](https://en.wikipedia.org/wiki/Network_science)

<sup>12</sup> Wikipedia, *Tree (Graph Theory)*, [https://en.wikipedia.org/wiki/Tree\\_\(graph\\_theory\)](https://en.wikipedia.org/wiki/Tree_(graph_theory))

<sup>13</sup> Wikipedia, *Set Theory*, [https://en.wikipedia.org/wiki/Set\\_theory](https://en.wikipedia.org/wiki/Set_theory)

<sup>14</sup> *Lorem Ipsum*, <https://www.lipsum.com/>

## Stocks and Flows

To understand some terminology clearly, I want to explicitly define these terms. Information can be grouped into two general buckets: stocks and flows.

A **stock** is information as of some specific point in time. The XBRL technical syntax uses the notion of the “instant” period type to represent a stock. A **flow** is information between two points in time. The XBRL technical syntax uses the notion of “duration” period type to represent a flow.

## Roll Up and Roll Forward

A **roll up** is simply an aggregation. You can aggregate a set of stocks or you can roll up a set of flows. But, you cannot roll up a mixture of stocks and flows.

A **roll forward** is simply a change between two stocks. A roll forward only applies between the relation of the same stock during some duration, or period, of time. You cannot have a roll forward of flows.

## Fragment

A **fragment** is simply a part of a report. Every report is made up of some set of fragments. The Lorem Ipsum report that I created has 11 fragments. The notion of a fragment can be confusing to people because of the many different ways of working with the XBRL building blocks used to create reports. XBRL provides both “Networks” and “Hypercubes” for representing fragments of a full report. I want to stay away of a complete discussion of notions of Networks, Hypercubes, and what I call a Block in order to keep this explanation of fragments simple<sup>15</sup>.

There is one final thing that is worth mentioning about fragments. Today, most public companies that are creating XBRL-based financial reports are doing so with a “presentation orientation” rather than a “representation of information orientation” and as such, they tend to create report fragments that are harder than necessary to understand. Keep this in mind if you look at XBRL-based financial reports to figure out the best way to represent information in your business reports.

## Business Rules

A **business rule** is simply a machine-readable representation of business logic. Every report is made up of some set of fragments of information that are used to organize facts that are represented in the report. Business rules help those creating such reports get the represented information correct and business rules also help those trying to make use of the reported facts to do so effectively and efficiently.

For example, in my Lorem Ipsum example I have a business rule that is represented using XBRL formation that states that “Ipsum = Lorem”.

#	Label	Result	Rule
1	The total amount of Ipsum must equal the total amount of Lorem. (REPORT_IpsumEqualsLorem)	Pass	\$Total_Ipsum = \$Total_Lorem
2	The total amount of Ipsum must equal the total amount of Lorem. (REPORT_IpsumEqualsLorem)	Pass	\$Total_Ipsum = \$Total_Lorem

This rule specifies that the total amount of Ipsum must always equal the total amount of Lorem. Roll ups have rules. Roll forwards have rules. Every roll up and roll forward

---

<sup>15</sup> If you are anxious to better understand networks, hypercubes, and blocks please see the document, *Putting the Expertise into an XBRL-based Knowledge Based System for Creating Financial Reports*, <http://pesseraact.azurewebsites.net/PuttingTheExpertiseIntoKnowledgeBasedSystem.pdf>

represented in a report must follow these business rules. Other types of business rules, or business logic, can also exist.

## Roll Up of Stocks

In the example XBRL-based report, fragment 11 shows a roll up of stocks:

Component: (Network and Table)		
Network	11 - Schedule - <b>Fragment 11</b>	
Table	Implied [Table]	
Reporting Entity [Axis]	30810137d58f76b84afd <a href="http://standards.iso.org/iso/17442">http://standards.iso.org/iso/17442</a>	
Unit [Axis]	USD	
	Period [Axis] ▼	
Implied [Line Items]	2018-12-31	2017-12-31
<b>Etiam [Roll Up]</b>		
Eu eleifend augue	600	700
Est aliquet ante	200	100
Nunc mattis aliquam	200	200
<b>Etiam</b>	<b>1,000</b>	<b>1,000</b>

The graph is more apparent when you look at the roll up relations of this fragment using the roll up business rules of this fragment:

Label	Rendered Value	Op	Reported Value	Calculated Value	Balance	Result	Name
▼ Implied [Line Items]							fro:ImpliedLineItems
▼ Etiam [Roll Up]							basic:EtiamRollUp
Eu eleifend augue	600	+	600				basic:EuEleifendAugue
Est aliquet ante	200	+	200				basic:EstAliquetAnte
Nunc mattis aliquam	200	+	200				basic:NuncMattisAliquam
<b>Etiam</b>	<b>1,000</b>		<b>1,000</b>	<b>1,000</b>		Verified	basic:Etiam

Label	Rendered Value	Op	Reported Value	Calculated Value	Balance	Result	Name
▼ Implied [Line Items]							fro:ImpliedLineItems
▼ Etiam [Roll Up]							basic:EtiamRollUp
Eu eleifend augue	700	+	700				basic:EuEleifendAugue
Est aliquet ante	100	+	100				basic:EstAliquetAnte
Nunc mattis aliquam	200	+	200				basic:NuncMattisAliquam
<b>Etiam</b>	<b>1,000</b>		<b>1,000</b>	<b>1,000</b>		Verified	basic:Etiam

## Roll Up of Flows

In the example XBRL-based report, fragment 02 shows a roll up of flows:

Component: (Network and Table)	
Network	02 - Schedule - Fragment 02
Table	Implied [Table]
Reporting Entity [Axis]	30810137d58f76b84afd <a href="http://standards.iso.org/iso/17442">http://standards.iso.org/iso/17442</a>
Unit [Axis]	USD
	Period [Axis] ▼
Implied [Line Items]	2018-01-01/2018-12-31
<b>Sollicitudin (Ultricies) [Roll Up]</b>	
<b>Pellentesque (Magna) [Roll Up]</b>	
Fermentum	4,000
Nunc	2,000
Pellentesque (Magna)	2,000
Euismod	0
Sollicitudin (Ultricies)	2,000

The graph is more apparent when you look at the roll up relations of this fragment using the roll up business rules of this fragment:

Label	Rendered Value	Op	Reported Value	Calculated Value	Balance	Result	Name
Implied [Line Items]							fro:ImpliedLineItems
Sollicitudin (Ultricies) [Roll Up]							basic:SollicitudinUltriciesRollUp
Pellentesque (Magna) [Roll Up]							basic:PellentesqueMagnaRollUp
Fermentum	4,000	+	4,000				basic:Fermentum
Nunc	2,000	-	2,000				basic:Nunc
Pellentesque (Magna)	2,000	+	2,000	2,000		Verified	basic:PellentesqueMagna
Euismod	0	-	0				basic:Euismod
Sollicitudin (Ultricies)	2,000		2,000	2,000		Verified	basic:SollicitudinUltricies

## Roll Forward

In the example XBRL-based report, fragment 04 shows a roll forward of a stock between two periods of time:

Component: (Network and Table)	
Network	04 - Schedule - Fragment 04
Table	Implied [Table]
Reporting Entity [Axis]	30810137d58f76b84afd <a href="http://standards.iso.org/iso/17442">http://standards.iso.org/iso/17442</a>
Unit [Axis]	USD
	Period [Axis] ▼
Implied [Line Items]	2018-01-01/2018-12-31
<b>Nulla [Roll Forward]</b>	
Nulla, Beginning Balance	3,000
Malesuada	3,000
Finibus euismod	(2,000)
Nullam	6,000
Semper	(1,000)
Egestas	(5,000)
Nulla, Ending Balance	4,000

The graph is more apparent when you look at the roll forward relations of this fragment using the business rules of this fragment:

Label	Rendered Value	Op	Reported Value	Calculated Value	Balance	Result	Name
Implied [Line Items]							fro:ImpliedLineItems
Nulla [Roll Forward]							basic:NullaRollForward
Nulla, Beginning Balance	3,000	+	3,000				basic:Nulla
Malesuada	3,000	+	3,000				basic:Malesuada
Finibus euismod	(2,000)	-	2,000				basic:FinibusEuismod
Nullam	6,000	+	6,000				basic:Nullam
Semper	(1,000)	-	1,000				basic:Semper
Egestas	(5,000)	-	5,000				basic:Egestas
Nulla, Ending Balance	4,000	+	4,000	4,000		Verified	basic:Nulla

## Adjustment (i.e. correction of an error)

In the example XBRL-based report, fragment XX shows an adjustment between two instants between two different reports for the same point in time: (i.e. the current report is correcting an error in the prior report)

COMING SOON!

## Business Report Semantics

Breaking the information into pieces you get the following:

- **Leaf nodes** that contain information that represent stocks and flows
- **Metadata** that represents the **aggregation logic** of the leaf nodes; that aggregation logic includes
  - **Roll ups** as of some *point in time* (stocks) or *during some period of time* (flows)
  - **Roll forwards** *across time* between two periods of time for a stock
  - **Adjustments** (no example is shown but this is known to exist) which documents a *change in value* (such as to correct an error).
- **Metadata** that represents the structure of information, mechanical relations, and other **business logic** of the information.

The supporting metadata allows both creators of information and consumers of information to be certain that they are working with high-quality information.

## Report Fragment Graph

As I have said, a report has some set of fragments that are reported. Business rules could exist to mandate when some fragment is required to be included within a business report:



**CC0 1.0 Universal (CC0 1.0)  
Public Domain Dedication**

CC0 1.0 Universal (CC0 1.0) Public Domain Dedication <https://creativecommons.org/publicdomain/zero/1.0/>

#	Disclosure	Category	Level	Pattern	Applicable	Found	Disclosure Consistent	Representation Concept [TEXT BLOCK]	Representation Concept [DETAIL]	Checklist Category	Reason
1	<a href="#">Balance Sheet</a>		Level4Detail	COMPONENT	True	True	CONSISTENT	NOT-EXPECTED	NOT-EXPECTED	Required disclosure	Disclosure always required, satisfied by Assets [Roll Up] and Liabilities and Equity [Roll Up]
2	<a href="#">Assets [Roll Up]</a>		Level4Detail	ROLL UP	True	True	CONSISTENT	NOT-EXPECTED	Assets	Part of disclosure	Disclosure always required
3	<a href="#">Liabilities and Equity [Roll Up]</a>		Level4Detail	ROLL UP	True	True	CONSISTENT	NOT-EXPECTED	Liabilities and Equity	Part of disclosure	Disclosure always required
4	<a href="#">Income Statement</a>		Level4Detail	ROLL UP	True	True	CONSISTENT	NOT-EXPECTED	Net Income (Loss)	Required disclosure	Disclosure always required
5	<a href="#">Cash Flow Statement, Direct Method</a>		Level4Detail	ROLL UP	True	True	CONSISTENT	NOT-EXPECTED	Net Cash Flow	Required disclosure	Disclosure always required
6	<a href="#">Receivables [Roll Forward]</a>		Level4Detail	ROLL FORWARD	True	True	CONSISTENT	NOT-EXPECTED	Receivables	Required disclosure	Disclosure always required
7	<a href="#">Cash and Cash Equivalents [Roll Forward]</a>		Level4Detail	ROLL FORWARD	True	True	CONSISTENT	NOT-EXPECTED	Cash and Cash Equivalents	Line item exists, then disclosure required	Required because line item basic: CashAndCashEquivalents was reported
8	<a href="#">Inventories [Roll Forward]</a>		Level4Detail	ROLL FORWARD	True	True	CONSISTENT	NOT-EXPECTED	Inventories	Line item exists, then disclosure required	Required because line item basic: Inventories was reported
9	<a href="#">Accounts Payable [Roll Forward]</a>		Level4Detail	ROLL FORWARD	True	True	CONSISTENT	NOT-EXPECTED	Accounts Payable	Line item exists, then disclosure required	Required because line item basic: AccountsPayable was reported
10	<a href="#">Property, Plant, and Equipment [Roll Forward]</a>		Level4Detail	ROLL FORWARD	True	True	CONSISTENT	NOT-EXPECTED	Property, Plant and Equipment	Line item exists, then disclosure required	Required because line item basic: PropertyPlantAndEquipment was reported
11	<a href="#">Long-Term Debt [Roll Forward]</a>		Level4Detail	ROLL FORWARD	True	True	CONSISTENT	NOT-EXPECTED	Long-term Debt	Line item exists, then disclosure required	Required because line item basic: LongtermDebt was reported
12	<a href="#">Retained Earnings [Roll Forward]</a>		Level4Detail	ROLL FORWARD	True	True	CONSISTENT	NOT-EXPECTED	Retained Earnings	Line item exists, then disclosure required	Required because line item basic: RetainedEarnings was reported

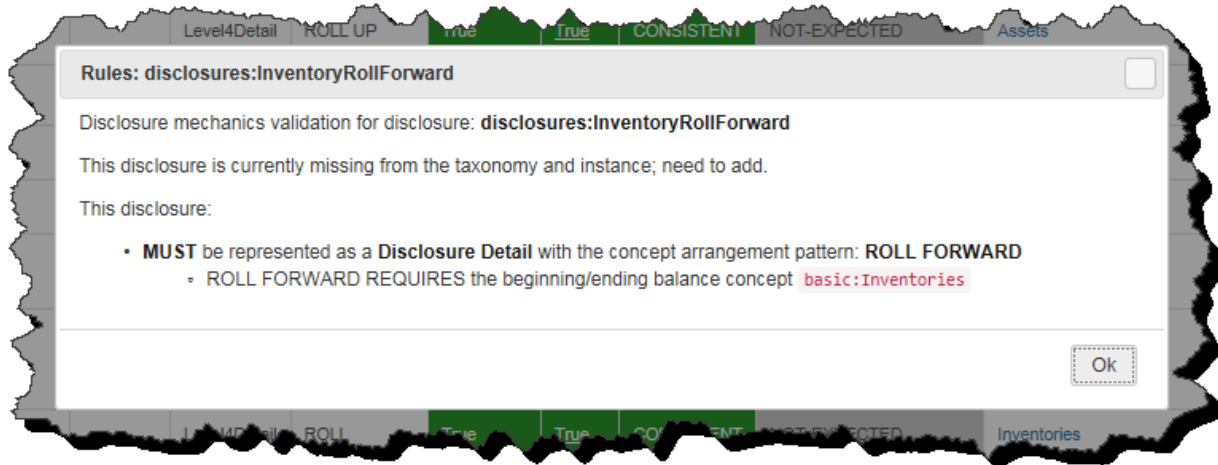
Two types of reporting checklist rules were used for this basic example. First, rules stating that specific fragments were required to be included within the report, *required fragments*. See lines 1 through 6 in the graphic above. Second, a rule was created for leaf node that was reported stating that leaf node is reported, then a roll forward of that leaf node is also required to be reported. See lines 7 through 12 in the graphic above. These reporting checklist rules were expressed using XBRL definition relations<sup>16</sup>.

## Fragment Mechanics Graph

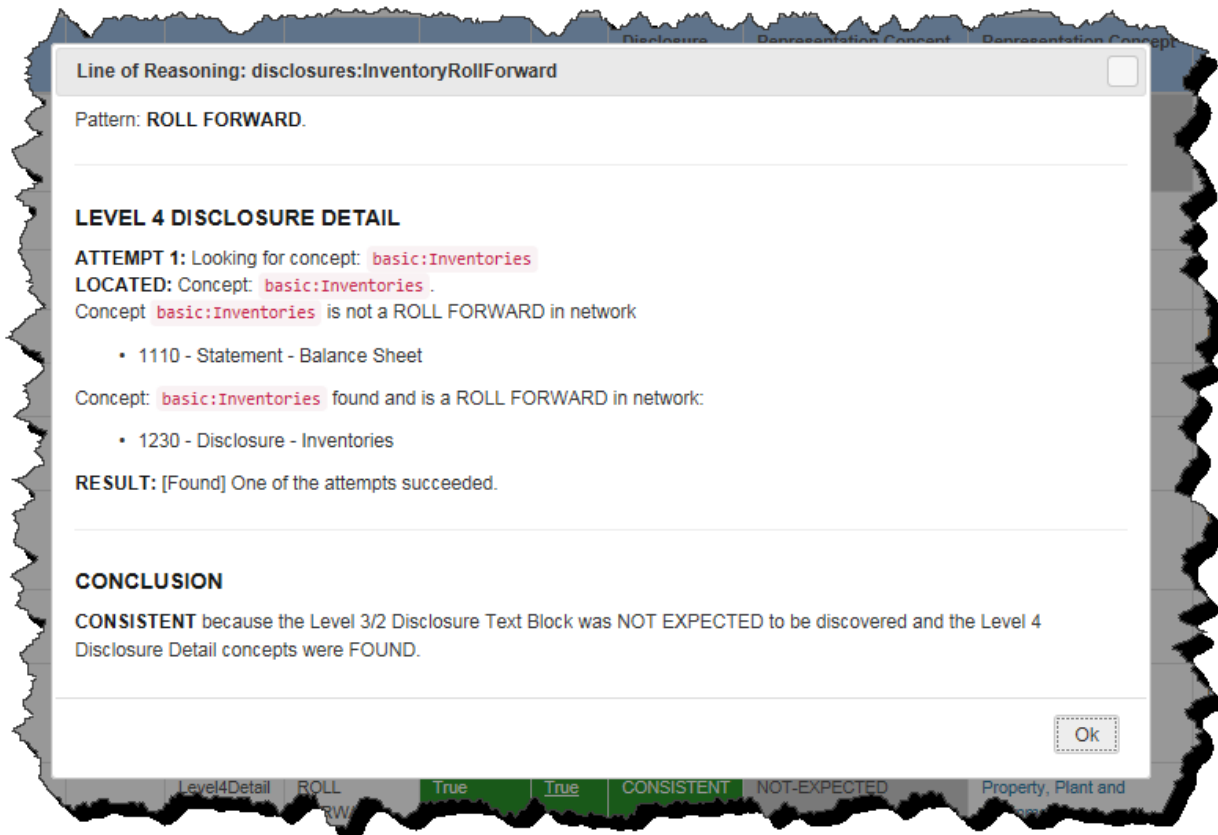
Business rules that represent the mechanics rules check to make sure that a fragment is represented correctly within a business report. Fragment mechanics rules test logic of the XBRL representation to make sure a fragment is consistent with what is expected. For example a specific report fragment which is a roll forward would require some specific concept to be used to represent the instants or stocks of that roll forward. The fragment mechanics rules are also represented using XBRL definition relations<sup>17</sup>.

<sup>16</sup> Reporting checklist rules expressed using XBRL definition relations, <http://xbrlsite.azurewebsites.net/2018/Prototypes/Basic/Basic-XASB-ConsistentRF/DisclosureMechanicsRules/ReportingChecklist-xasb-rules-def.xml>

<sup>17</sup> Disclosure mechanics rules for all disclosures, <http://xbrlsite.azurewebsites.net/2018/Prototypes/Basic/Basic-XASB-ConsistentRF/DisclosureMechanicsRules/disclosure-mechanics-xasb.xsd>



A second use of the fragment mechanics rules is for identifying the report fragment within the actual report.



Essentially, what the Line of Reasoning is saying is that it found the concept “basic:Inventories” within the balance sheet, but it was not a roll forward and so that network was not used. The software continued searching and found the concept “basic:Inventories” again, this time it was within a roll forward, no other instances of this concept were identified and therefore the software reached the conclusion that this must be the “Inventories Roll Forward” disclosure.

## Consistency Cross-checks Graph

Another type of relation that exists within a business report is what is called the consistency cross-check. Consistency cross-check relations are used to make sure reported facts are consistent with the expected relations with other reported facts.

#	Label	Result	Rule
1	CashFlowNet = CashFlowProvidedByUsedInOperatingActivitiesNet + CashFlowsProvidedByUsedInInvestingActivitiesNet + CashFlowsProvidedByUsedInFinancingActivitiesNet (FAC_CF1_NetCashFlowEqualsOperatingInvestingAndFinancing)	Pass	\$CashFlowNet = \$CashFlowProvidedByUsedInOperatingActivitiesNet + \$CashFlowsProvidedByUsedInInvestingActivitiesNet + \$CashFlowsProvidedByUsedInFinancingActivitiesNet
2	Liabilities = Current Liabilities + Noncurrent Liabilities (FAC_BS3_LiabilitiesEqualsCurrentPlusNoncurrentLiabilities)	Pass	\$Liabilities = \$LiabilitiesCurrent + \$LiabilitiesNoncurrent
3	Liabilities = Current Liabilities + Noncurrent Liabilities (FAC_BS3_LiabilitiesEqualsCurrentPlusNoncurrentLiabilities)	Pass	\$Liabilities = \$LiabilitiesCurrent + \$LiabilitiesNoncurrent
4	LiabilitiesAndEquity = Liabilities + Equity (FAC_BS4_LiabilitiesAndEquityEqualsLiabilitiesPlusEquity)	Pass	\$LiabilitiesAndEquity = \$Liabilities + \$Equity
5	LiabilitiesAndEquity = Liabilities + Equity (FAC_BS4_LiabilitiesAndEquityEqualsLiabilitiesPlusEquity)	Pass	\$LiabilitiesAndEquity = \$Liabilities + \$Equity
6	Assets = Current assets + Noncurrent assets (FAC_BS2_AssetsEqualsCurrentPlusNoncurrentAssets)	Pass	\$Assets = \$AssetsCurrent + \$AssetsNoncurrent
7	Assets = Current assets + Noncurrent assets (FAC_BS2_AssetsEqualsCurrentPlusNoncurrentAssets)	Pass	\$Assets = \$AssetsCurrent + \$AssetsNoncurrent
8	GrossProfitLoss = RevenuesNet - CostOfSales (FAC_IS1_GrossProfitFoots)	Pass	\$GrossProfitLoss = (\$RevenuesNet - \$CostOfSales)
9	Assets = Liabilities and Equity (FAC_BS1_BalanceSheetBalances)	Pass	\$Assets = \$LiabilitiesAndEquity
10	Assets = Liabilities and Equity (FAC_BS1_BalanceSheetBalances)	Pass	\$Assets = \$LiabilitiesAndEquity

## Type or Class Relations Graph

Another type of relation that exists within a report is what is called the type or class relation. Type or class relations are similar to consistency cross-check relations except that they are not limited to mathematical relations. Type or class relations make sure that, say, a concept is not inadvertently used within the tree of relationships that are illogical. For example, the concept “basic:PellentesqueMagna” should never be part of “basic:Fermentum”.

No example of type or class relations is provided because the software vendor providing these examples does not support this sort of relation at this time.

## Basic Disclosures Patterns Followed by All Other Disclosures

While this example was created to communicate specific information as straight-forward and simply as possible; don’t be fooled by the basic nature of this example. There are additional relation patterns and additional fragments that have been analyzed<sup>18</sup>.

## More Formal Mathematical Representation of Graphs

As stated, this document provides an information summary of the graphs of relations within a business report. An improvement on this informal summary would be a formal summary in mathematical terms. As I am not a mathematician, I do not have the expertise to create such a formal summary.

## For More Information

For more information about XBRL-based financial reports and how general business reports can leverage ideas used in financial reporting, please see the document *Intelligent XBRL-based Digital Financial Reporting*<sup>19</sup>.

<sup>18</sup> Best practice examples of 65 disclosures, <http://www.xbrl.com/site1/2017/Prototypes/DisclosureAnalysis/All/>

For additional general examples that are significantly more comprehensive but relate to financial reporting, please see the document *Accounting Process Automation Using XBRL*<sup>20</sup>. Also, see the document *Guide to Implementing Robotic Finance*<sup>21</sup> for additional details.

## Technical Information

This section contains helpful technical information for those interested in understanding the scheme used to implement XBRL-based accounting process automation, financial report creation automation, or other such process automation.

All the XBRL used in these examples has been validated by four separate XBRL processors. All XBRL has been loaded and processed using two completely separate implementations of this scheme, one of which is commercially available for purchase today.

### **Run the Basic Example using XBRL Cloud (shown in this document) or the Pesseract application**

To duplicate the processing of this example using XBRL Cloud, do the following.

1. Load this RAW XBRL instance, **NO XBRL Formulas or other rules**:  
[http://xbrl.azurewebsites.net/2018/Prototypes/LoremIpsum/basic-SampleInstance\\_NoFormulas.xml](http://xbrl.azurewebsites.net/2018/Prototypes/LoremIpsum/basic-SampleInstance_NoFormulas.xml)
2. Load this RAW XBRL instance, WITH XBRL Formulas:  
[http://xbrl.azurewebsites.net/2018/Prototypes/LoremIpsum/basic-SampleInstance\\_WithFormulas.xml](http://xbrl.azurewebsites.net/2018/Prototypes/LoremIpsum/basic-SampleInstance_WithFormulas.xml)
3. Load this RAW XBRL instance, WITH XBRL Formulas and WITH all other rules: (Coming soon)
4. Load this Inline XBRL instance:  
[http://xbrl.azurewebsites.net/2018/Prototypes/LoremIpsum/basic-SampleInstance\\_WithFormulas.html](http://xbrl.azurewebsites.net/2018/Prototypes/LoremIpsum/basic-SampleInstance_WithFormulas.html)
5. Use this schema for reporting checklist and disclosure mechanics validation: (Coming soon)
6. Report fragments and business rules results: (Coming soon)
7. Reporting checklist and disclosure mechanics results: (Coming soon)

---

<sup>19</sup> Charles Hoffman and Rene van Egmond, *Intelligent XBRL-based Digital Financial Reporting*, <http://xbrl.squarespace.com/intelligent-xbrl/>

<sup>20</sup> Charles Hoffman, *Accounting Process Automation Using XBRL*,  
<http://xbrl.azurewebsites.net/2018/Library/AccountingProcessAutomationUsingXBRL.pdf>

<sup>21</sup> Charles Hoffman, *Guide to Implementing Robotic Finance*,  
<http://xbrl.azurewebsites.net/2018/RoboticFinance/GuideToImplementingRoboticFinance.pdf>