Financial Report Creation Application (Prototype)

By Charles Hoffman, CPA

This document summarizes ideas related to creating a software application which specializes in creating financial reports. The application is not like Microsoft Word which does not understand financial reports, rather the application leverages the structure and semantics of the XBRL format and has an intimate understanding of financial reports.

Main interface:

Main interface shows the components of a financial report on the left and sections or “folders” which contain information which helps organize these components on the right. For example, the components of a financial report can be broken down into the categories: primary financial statements, basis of reporting, disclosures, supplementary financial information, and so forth:

Users of the application determine the categories they desire to use to organize their financial report components.
Search through components:

The application user can search through the list of financial statement components to find the component with which they would like to work. For example, if the user types “Debt” in the search box (see upper right), all the components which have the term “debt” in them appear in the search results listed on the left as a flat list.

Two things are highlighted here: the notion of components and the notion of organization. Financial reports have an ordering or sequence or flow of the financial report. The application manages this using standard metadata and metadata provided by the user.
Workflow management:

The application captures information from what the user does, information the user enters, and algorithms within the software such as validation of the information to manage workflow. Below you can see a screen shot which shows the status of the primary financial statement category of components; you can see the status, validation results, a progress bar which shows how much more work is anticipated, and the creation date of each component. Users can configure which information which they might like to see in their workflow view.
Below you can see that the workflow information leverages the hierarchy of components. If the user selects the entire document, the workflow management information for the sections rather than for each component are shown.

Collaboration

Some versions of a financial report creation tool might be used only by a single user. Other tools may need to be collaborative (http://xbrl.squarespace.com/journal/2011/10/6/creation-of-financial-statement-is-a-collaboration.html)
Closing Book:

The financial statement creation application is not just for creating the financial statement; but rather it also manages the “closing book” which contains the schedules and other information which supports and details the financial report. These could be internal schedules or external auditor lead schedules. Today, this information is generally managed within Excel spreadsheets. This application would also use spreadsheet-type components to contain this information but those spreadsheets are not presentation based artifacts with columns “A, B, and C” and rows “1, 2, and 3”; but rather these are semantic spreadsheets which have connections to the summary information within the financial report.

Further, connections can also be created to information external to the application such as queries to a database, etc.
Editing component information:

Information for each component is edited in the semantic rendering view (or the presentation rendering view perhaps), what you see is what you get. Key numbers in, copy/paste from Excel, or tie information to the closing book (i.e. edit the closing book, not the actual financial).
Semantic renderings are dynamic:

The semantic renderings (the actual information on the right) are easily read by humans and adjustable like a pivot table if you want a different view of the information.
There is no difference between how text and numbers are processed, they all fit into the same logical model.
**Quantrix Modeler**

The Quantrix modeler is an example of a model based financial modeling tool. This shows you how a user would interact with each component. Quantrix calls components a “matrix”. The demos and tutorial give you a really good idea how this could work:

Demos: [http://www.quantrix.com/Quantrix_Video_Demos.htm](http://www.quantrix.com/Quantrix_Video_Demos.htm) (Top two videos are best)

Tutorial: [http://www.quantrix.com/tutorial/tutorial.php](http://www.quantrix.com/tutorial/tutorial.php) (This is 30 minutes, but worth the time)

The Quantrix modeler does not support text, therefore it will not work as is for financial reports which contain text. But, the tools which are trying to create XBRL today will eventually (a) work in this manner and (b) support numbers and text. Further, the Quantrix format is proprietary to Quantrix and you must have the Quantrix tool to work with them or view them, unless you export the information to another proprietary tool; Excel. But, with XBRL there is no one proprietary tool, there are many, many tools which each support the XBRL global standard.
iTunes-type navigation/Interface:

Imagine an iTunes-type interface for organizing, creating, and working with a financial statement.


The above is the THIRD option of the “View”. Can see an image of the rendering, descriptive information.
Above is still the THIRD view, but with the image expanded and therefore less of the list showing

Above is the SECOND “View” option, focusing on the images.

Note the SEARCH box, where you can select what you would search by selecting the check box

Above is the FIRST option, focusing on the list which can be easily filtered using the three boxes on the top.
On the side you can organize the “filings” of a reporting entity within a library, like the play lists in iTunes. You can create folders like “Current Projects” and “Favorites” and put play lists into the folders.

Note the “Store” where you can connect to the “iTunes Store” or the Edgar Online Store and purchase XBRL Formulas, financial ratios, etc.

The “PLAYLISTS” is like a personal database or set of queries. The actual filings would be in the “LIBRARY” under “Filings.”
iPad type financial report reader:

Imagine an iPad-type financial report reader. This San Francisco Chronicle new paper reader can show you what something like that might look like:


This is a newspaper, but image this being a financial report:

Rather than seeing the sections of a newspaper, you see the sections of a financial report:
Navigating from section to section of a financial report:
Scratch

Scratch is an application for elementary and middle school students. The application allows them to program complex animations. Scratch was created by MIT, see http://scratch.mit.edu/.

The important notion which is shown by Scratch is that of “building blocks”. The building block are interconnected in only appropriate ways; inappropriate interconnections are not allowed because they are not logical. The software manages what is logical and what is not based on rules that the software application understands.

The notion of a “radically tailorable software application” was summarized in a blog post:


Business users don’t write the complex rules using complex technical syntax and other stuff which they don’t understand which are used to specify the building blocks; business users interact at the level of using the building blocks as they are allowed to be used. Other more skilled business users work with technical people, if necessary, to create or modify building blocks.

GoAnimate is another application which shows the power of building blocks:

Interactive information viewer:

This rendering is similar to the SEC interactive data viewer. Note that this rendering is a prototype, does not render all information patterns correctly due to my lack of programming skills, but (a) could look WAY better than this with NO additional information added to the XBRL; and (b) if additional information was added, could look closer to what a Word document of a financial statement looks like.

Streamed rendering like today’s paper reports:

The renderings can be “streamed” together to generate a complete financial statement. This is a prototype of streaming the pieces together, the rendering would be WAY better than this and much closer to what a Word document looks like today:


Imagine the above rendering, but try to project the following into the rendering:

- The rendering could be in HTML, Word, PDF, Excel, or in any other format
- Formatting characteristics are more like common SEC-type 10-K and 10-Q type formatting, not the highly colorized version on the prototype which was formatted for some other purpose
- The rendering engine understood the rendering models for the information metapatterns: roll up, roll forward, adjustment, variance, etc.
- Other formatting adjustments to make it how someone wants the information rendered, options for as much as is reasonable; No element names are used at all, all labels; no namespaces or other technical junk is shown

A “pixel perfect” rendering could be created, but that would mean a one-to-one mapping from a financial report to the preferred rendering format of a specific financial report creator or consumer. But, you can get a standard rendering which is very, very usable, may not be exactly what you want, but it would work well and you don’t have to do ANY work, the rendering engine understands everything it needs to understand from the metapatterns and the XBRL instance and taxonomy. This is why XBRL separated semantic meaning and presentation; it is easy to agree on semantic meaning, less likely that presentation format would ever be agreed to.

Basically, the rendering would look just like a financial report today, you could change some parameters and modify the look of the rendering, but you could not control 100% of the rendering, unless you chose
to create a one-to-one rendering which you would likely choose not to do because the automated rendering is very satisfactory.