



# XBRL SOFTWARE REQUIREMENTS

Considerations for Development of  
XBRL Taxonomy Editor and Instance  
Document Creator Software

**May 2003**

AICPA XBRL Implementation Task Force

## **AICPA XBRL Implementation Task Force:**

**Members:** American Institute of Certified Public Accountants (AICPA)  
Ernst & Young LLP (Ernst & Young)  
KPMG LLP (KPMG)  
PricewaterhouseCoopers LLP (PwC)

**Chair:** Paul Penler, Principal, Ernst & Young LLP

**Contact:** Jeff Naumann, AICPA, [jnaumann@aicpa.org](mailto:jnaumann@aicpa.org)

The XBRL Software Requirements Guide was developed by the members of the AICPA XBRL Implementation Task Force to benefit software developers who are incorporating XBRL functionality into their products. The non-authoritative requirements in this Guide offer alternatives that software developers can choose from and then modify, as necessary, to meet their needs. They are not intended to establish standards or preferred practices.

Copyright 2003 by  
American Institute of Certified Public Accountants, Inc.  
New York, NY 10036-8775

## Table of Contents

<b>Introduction</b>	<b>3</b>
<b>Taxonomy Editor Requirements</b>	<b>3</b>
Minimum Requirements ( <i>Must Have</i> )	<b>3</b>
Suggested Requirements ( <i>Nice to Have</i> )	<b>5</b>
Optional Requirements	<b>7</b>
<b>Instance Document Creator Requirements</b>	<b>8</b>
Types of Instance Document Creators Used	<b>8</b>
Minimum Requirements ( <i>Must Have</i> )	<b>9</b>
Suggested Requirements ( <i>Nice to Have</i> )	<b>10</b>
Optional Requirements	<b>12</b>
<b>Additional Findings</b>	<b>12</b>
Description of Process Improvement Tool(s)	<b>12</b>
Instance Document Creator/Taxonomy Editor	<b>13</b>
Automatic Instance Document Creator/Editor	<b>13</b>
<b>Conclusion and Wrap Up</b>	<b>14</b>

# Introduction

This document is being provided as a summary to the software industry of the findings from the XBRL Implementation Task Force. The XBRL Implementation Task Force was sponsored by the AICPA and involved the professional services firms of Ernst & Young LLP, KPMG LLP and PriceWaterhouseCoopers LLP.

The task force's project goal was to exercise the entire process of creating instance documents from full financial statements. Each audit firm selected clients and obtained full financial statements from those clients. Auditors from each firm mapped every fact in the statements to the US-GAAP-CI taxonomy and noted facts that did not fit. Company specific extension taxonomies were created using the information regarding the facts that did not match concepts in the US GAAP CI taxonomy. It was also decided to create a taxonomy for tagging the notes to the financial statement with greater detail than the US GAAP CI taxonomy allowed. The task force then used the US GAAP CI taxonomy, the extension taxonomies and the mapping information to create the instance documents for each company.

The types of software used in this project were XBRL taxonomy editors and instance document creators. There was strong support from all vendors that participated and from the experiences of this project came a set of general requirements for these two types of XBRL software.

## Taxonomy Editor Requirements

The XBRL Implementation Task Force project required the creation of extension taxonomies to fully represent the financial statements of the selected companies. However this document also includes the requirements that would be necessary for the creation of the base taxonomies.

### Minimum Requirements *(Must Have)*

The "Minimum Requirements" is a suggested list of requirements necessary to create a successful XBRL taxonomy editor. This is not meant to be an exhaustive list, but a helpful collection of the requirements the project identified as being essential for an XBRL taxonomy editor.

- TE NR1.** Must create a valid XML Schema File that adheres to the XML Schema definition schema as well as the XBRL definition schema.
- TE NR2.** Must create valid linkbases for Label, Presentation, Calculation, Reference, and Definition. The linkbases must use the correct syntax for XLink and must have valid XPointer locations. *(Future tools will need to create Rulebase and Versioning linkbases, when the specifications for these linkbases are finalized.)*
- TE NR3.** All links created should have all attributes that have been defined in the XBRL linkbase schema, including priority. All link attributes should be editable; however, attributes that typically would not be changed by a user should be hard to intentionally change.
- TE NR4.** Must use XML Schema "import" feature to allow imported taxonomies.
- TE NR5.** Must have the ability to create elements with all available data types defined in the XBRL schema.
- TE NR6.** Must have the ability to create abstract elements.

- TE NR7.** Must have the ability to create tuples and tuple contents.
- TE NR8.** Must have the ability to denote balance type (Debit, Credit, None) for items.
- TE NR9.** Must have the ability to create, edit, delete and override labels.
- TE NR10.** Must have the ability to create and edit element names.
- TE NR11.** Must have the ability to specify different languages and roles for label links.
- TE NR12.** Must have the ability to create multiple label roles.
- TE NR13.** Must have the ability to view and edit element descriptions and include those descriptions in the documentation element under annotation for the element.
- TE NR14.** Ability to create dimension-element links, often referred to as "Same-As" links. Ability to create new link types as defined by XBRL 2.1.
- TE NR15.** Must include the ability to create enumerations.
- TE NR16.** Must include the ability to customize the fields for the reference linkbase. In other words, define the breakdown for and then view and edit reference links, and maintain multiple reference links for each element.
- TE NR17.** Must be able to add, change and delete calculation weights and hierarchical relationships for Calculation links.
- TE NR18.** Must be able to set, change and delete presentation order and hierarchical relationships on Presentation links.
- TE NR19.** Must be able to prohibit any link in all linkbases. Should visually display that prohibited links have been prohibited. This means that the tool must differentiate between the taxonomy being worked on specifically and imported taxonomies.
- TE NR20.** Must have the ability to create and edit the namespace and namespace identifier for the taxonomy. Also should be able to create and edit the names used for all saved files.
- TE NR21.** Must have the ability to search the elements both by element name and by label. The label search should search all label roles, not only the standard label role.
- TE NR22.** Must have the ability to update a taxonomy from a previous version of the XBRL specification to a new version as they are released. For example, if a taxonomy is created using the latest specification (2.0a) and then a new version comes out, the tool should be able to automatically update it to the new specification.
- TE NR23.** Must have the ability to choose a default label role to work with.
- TE NR24.** Must include basic reporting.
- TE NR25.** Must include help documentation or a detailed user manual.
- TE NR26.** Must be able to open a valid XBRL taxonomy including linkbases, from other vendor tools, as well as from its own output. A valid taxonomy may not have certain linkbases, so a tool should not error if an expected linkbase is not referenced in the schema or included.
- TE NR27.** Must have validation for schema and linkbases on save and on load/import.
- TE NR28.** Must be mostly self-installing.

- TE NR29.** Must have acceptable processing time. This is necessary for an optimal user experience and efficiency.
- TE NR30.** Must have a reasonable footprint on the users' machines. Can not require tremendous amounts of memory.

## Suggested Requirements (*Nice to Have*)

The "Suggested Requirements" are recommended to be a part of a user friendly or superior taxonomy editor. These are not necessary for a taxonomy editor to be functional; however, these requirements are highly recommended for successful taxonomy software.

- TE SR1.** Need a feature that imports a list of new elements and related information from Excel, CSV or other formats. The import should allow all properties of a new element to be input at once. All possible linkbase information should also be included (labels or references for example).
- TE SR2.** Ability to create links for multiple linkbases at one time, or to specify to only create a link for one linkbase at a time. Would need a way to choose which linkbases automatically get created and which don't.
- TE SR3.** Ability to easily set the properties of links on link creation. For example: label role, calculation weights, etc.
- TE SR4.** Option to remove certain words from element names. This would eliminate a list of common words not usually intended to be in the element name (a, and, the). Automated creation (optional) of element names and ids from label entry.
- TE SR5.** Copy or cut and paste of elements and linkbase items (presentation, calculation, labels). Should also be able to copy sections of hierarchies and paste within the same linkbase or add to another linkbase.
- TE SR6.** Ability to rename a group of (or entire set of) element names based on certain criteria.
- TE SR7.** An undo feature to allow a user to undo the last action. Would be more valuable if it could remember the last few actions.
- TE SR8.** Include multi select actions for common actions such as editing properties. User chooses many elements at once and then performs the same action to all at once, instead of having to go through one by one.
- TE SR9.** The search feature should find each instance of the element in a visual display of the linkbases.
- TE SR10.** Ability to spell check element names, labels and documentation.
- TE SR11.** Advanced functionality/user interface for creating new label roles (e.g. Use an existing role as the basis for a new role, display sortable flat list of labels side by side, etc)
- TE SR12.** Ability to move elements from one schema file to another and update linkbases accordingly. This needs to be highly controlled so a user would not accidentally use this feature.
- TE SR13.** Ability to sort and filter the list of elements on many different criteria. (i.e. item type, property values or linkbases.)
- TE SR14.** Good visual display of linkbases. (Example: Collapsible tree view for presentation links.)

- TE SR15.** Intuitive user interface with easy to access features. Examples include floating toolbars or right click menus.
- TE SR16.** Ability to view or not to view prohibited links in the visual display of the linkbases.
- TE SR17.** Ability to see the namespace prefix with each element, or an equivalent way to easily differentiate between namespaces and determine what namespace an element is from.
- TE SR18.** Ability to change the ordering for the listing of elements. Users might have different preferences or different needs.
- TE SR19.** Ability to see parent-child relationships for the current element selected in the tool. For example, a user selects an element name and they should be able to see all of the parent-child relationships in the different linkbases (presentation, calculation, definition). Should also be able to see labels and reference links.
- TE SR20.** Should visually see the difference between different element types (Item, abstract, tuple).
- TE SR21.** Ability to visually notify the user that elements are not included in a linkbase.
- TE SR22.** An XML view that displays the schema and linkbases from within the tool.
- TE SR23.** Advanced user interface for working with complex relationships (e.g. table view — Statement of Equity). A tree view is not sufficient for working with multi-dimensional relationships.
- TE SR24.** Ability for a user interface to display an alternate label if there is no entry for the default label role chosen.
- TE SR25.** Option to view a taxonomy with imported elements with or without imported elements.
- TE SR26.** Ability to view/report all elements with no documentation.
- TE SR27.** Ability to view/report all elements with missing labels for a particular role.
- TE SR28.** Suitable reporting to display all items, all properties, all linkbases, sorts, filters, etc.
- TE SR29.** Ability to choose whether or not to see prohibited links in certain reports.
- TE SR30.** Notification in a report that elements appear multiple times in a linkbase (i.e. an element is a child of multiple elements in a visual hierarchy of the linkbase; therefore it would appear in the hierarchy multiple times). For example, a flag is present next to an element to alert a user that the same element is shown in the linkbase multiple times. This will assist someone mapping data to XBRL elements.
- TE SR31.** Shareable HTML or Excel hierarchical view of taxonomies for taxonomy review by less technical folk. UML like view for more technical types.
- TE SR32.** Option to check elements not to print. For example, some elements might have been inserted into the taxonomy as headers and a user might not want those items to appear in a report. A toggle switch for all elements would be handy — a user could turn items on/off for printing.
- TE SR33.** Ability to fix non-fatal errors on loading a taxonomy or on saving a taxonomy. Trap errors and allow a user to fix each error in a logical manner.
- TE SR34.** Validation on user input as it is entered. For example, if a user types in a non-valid XML character in the element name, the tool should catch this at the time of entry.

- TE SR35.** Ability to validate certain 'logical' taxonomy relationships and flag for user follow-up (e.g. calculation links on string items, calculation arcs with weights of zero, calculations between different data types, etc.)
- TE SR36.** Fast processing time. No lag time on user actions, minimal wait on loading and saving taxonomies.

## Optional Requirements

Items in the "Optional Requirements" list are intended to be extra features or requirements that could be useful for certain cases or that might not be necessary for all taxonomy software.

- TE AR1.** Wizards to lead designers through steps of taxonomy creation process.
- TE AR2.** Full XML Schema savvy to do restrictions, create complexTypes.
- TE AR3.** All on screen labels external to the programming so they can easily be changed to other languages. (Globalization)
- TE AR4.** Password protection and audit trail of changes.
- TE AR5.** Auto-save of work in process.
- TE AR6.** Report designer feature, to allow a user to completely design reports.
- TE AR7.** Integration with Instance Document creation and mapping tools. (See "Instance Document Creator/Taxonomy Editor" under "Additional Findings".)
- TE AR8.** Instance creation template automatically created from taxonomy tool.
- TE AR9.** Integration with XML database/repository for library functions, version control.
- TE AR10.** Web service access to taxonomies embedded in system/repository.
- TE AR11.** Vendor service to catalogue taxonomies available around the world for easy access and reuse.
- TE AR12.** Optional internal database for files to go with direct access to schemas to allow for protection of taxonomies in process.
- TE AR13.** Tools to assist in taxonomy collaboration work flow, including reporting on status, element level notes and comments, and more.
- TE AR14.** Tools to assist in comparing a new version of a taxonomy with a prior version; especially necessary if a custom taxonomy has been built on top of a standard taxonomy and the standard taxonomy has been corrected/updated.



# Instance Document Creator Requirements

## Types of Instance Document Creators Used

There are many types of instance document creators available and many more are being developed that will be released in the near future.

The main categories of instance document creators are:

**1. “Stand alone” instance document creators**

Software specifically geared towards manually entering data into a format that creates an XBRL instance document.

- a. Caseware (Usually not used as a “Stand alone”; however, that is how it was used in this project.)
- b. Corel XMetaL
- c. Fujitsu
- d. Semansys

**2. “ERP and General Ledger type” instance document creators**

This is generally a reporting feature of an ERP product that allows fields to be mapped to the elements in a taxonomy.

- a. SAP
- b. PeopleSoft

**3. “Database-type” instance document creators**

This type is typically used to map a database to a schema and create instance documents.

- a. Software AG
- b. IWAY

“Stand alone” instance document creators are not usually tied into a database. They are tools that you manually enter data into. The data is placed in fields that are mapped to a taxonomy you have loaded into the instance document creator. The fields are usually grouped in some manner to denote the context they are related to. The software user also creates the necessary contexts to be used.

“ERP and General Ledger type” instance document creators are typically an extension of the reporting capabilities of an ERP system, such as SAP. A mapping is predefined between the reporting fields in your ERP system and the elements in an XBRL taxonomy. When the user is ready to create the instance document, they choose an option to print XBRL or save XBRL. This action creates an instance document using the mapping previously created.

“Database-type” instance document creators allow the mapping of a relational database, or other data source, to an XBRL taxonomy. The function is then very similar to the ERP system process. After a mapping is created, a user can automatically create instance documents or create schedules to automatically generate instance documents at certain times.

## Minimum Requirements *(Must Have)*

The "Minimum Requirements" are the minimal requirements necessary to compose a successful XBRL instance document creator. This is not meant to be an exhaustive list, but a helpful collection of the requirements the project identified as being essential for an XBRL instance document creator.

- ID NR 1.** Create a valid XBRL instance document that is compliant with the latest XBRL specification.
- ID NR 2.** Ability to create and associate contexts with each value or establish columns of context against which to enter data. Contexts should be fully configurable by the user. Need to allow the user to set the following items:
  - 1. Context Attributes: ID (must be valid XML element name), Precision, CWA
  - 2. Numeric Context sub elements: <Unit> with sub element <Measure> that holds the unit of measurement with optional dual measurements with operators.
  - 3. All contexts sub elements: <Entity>, sub element <Identifier> with attribute Scheme, Sub element <Segment>, Optional sub element <Scenario>. <Period> with available sub elements of <StartDate>, <EndDate>, <Duration>, <Instant> and <Forever>.
  - 4. For more detail, see XBRL specification 2.0 or 2.0 a.
- ID NR 3.** Ability to handle tuples (multiple, record oriented instances of the same sets of elements with different content).
- ID NR 4.** Ability to create footnotes as defined in the Specification.
- ID NR 5.** The instance creator tool needs to be able to read the enumerations and aid the user by suggesting and limiting entries to those enumerated values.
- ID NR 6.** The user interface must help the user by warning of illegal characters, such as "&", " ", " ", " ", " ", " ", "<" and ">".
- ID NR 7.** Ability to choose the taxonomy files to use for creating an instance document. Tool should automatically import all taxonomies imported in the chosen taxonomies.
- ID NR 8.** Ability to search element labels, element names, definitions and data.
- ID NR 9.** Drag and drop of individual taxonomy elements or groupings of elements from a separate hierarchical view; use standard windows techniques (Ctrl\_shift, Ctrl Click) to pick ranges or multiple elements to bring into instance.
- ID NR 10.** Instance document creation tools should provide for the ability to include (display) or exclude (hide) values in an instance document that were reported in their originally formatted source as zero or empty.
- ID NR 11.** Ability to create standard templates for manual or automated entry.
- ID NR 12.** Ability to reorder and remove elements from instance creation screen with "Are you sure?" protection.
- ID NR 13.** Ability to view and print a completed instance document as XML and a more readable format.
- ID NR 14.** Ability to view taxonomy elements by linkbase hierarchies (presentation, calculation or definition) or alphabetically with namespace identification; with mouseOver or another tool to view definition in documentation or references from reference links.

- ID NR 15.** Software should provide for the ability to display balance (debit/credit) and weight information from the calculation linkbase, in order to give guidance when determining values as positives or negatives in instance documents. This option should display the weight information in an easily viewable location.
- ID NR 16.** Ability to display link base information. (Labels, presentation hierarchy, calculation hierarchy, definition hierarchy)
- ID NR 17.** Ability to view definitions/documentation of the elements.
- ID NR 18.** Ability to view labels or element names. Should be able to specify label role and language.
- ID NR 19.** Tools must be able to import/open a valid XBRL instance document.
- ID NR 20.** Validation of tagged data (instance document).
  - 1. Well-formed XML
  - 2. Valid XML
  - 3. Valid XBRL
- ID NR 21.** Must validate all entered data on save and open. This would check the data entered against the data type for the tagged element and for the context entered.
- ID NR 22.** Must have a reasonable footprint on the users' machines. Can not require tremendous amounts of memory.
- ID NR 23.** Must have acceptable processing time. This is necessary for an optimal user experience and efficiency.

## Suggested Requirements *(Nice to Have)*

The "Suggested Requirements" are provided as a recommendation for useful, user friendly or superior instance document creators. These are not necessary for an instance document creator to be functional, however; these requirements are highly recommended.

- ID SR 1.** Ability to accept drag and drop data.
- ID SR 2.** Ability to embed XML comments in the instance document related to each element or as a whole.
- ID SR 3.** Table view (like XML Spy) for entering tuple/table information.
- ID SR 4.** Excel/Access like entry columns/tables for easier copy/paste of multiple rows of information.
- ID SR 5.** Ability to drag and drop sections from existing instance documents for reuse in other documents.
- ID SR 6.** Ability to import (PDF, Word or Excel etc.) documents directly into the ID creator for tagging.
- ID SR 7.** Create mappings from spreadsheets and databases to taxonomies/templates for automated instance creation.
- ID SR 8.** Ability to tag non-contiguous data tables with different contexts. When importing source data in an existing table format as text into the instance document creation tool it becomes necessary to tag specific combinations of columns from the source table into different contexts. An example from a financial statement would be the Cash Flows section.

- ID SR 9.** Ability to map the same data value to multiple elements in the same taxonomy as well as elements in multiple taxonomies.
- ID SR 10.** Ability to highlight specific text within a given portion of tagged textual data and allow for its detailed tagging. For example, if a paragraph is imported into the tool, a user should have the ability to select different portions of the text to uniquely tag. This would allow for the detail tagging of a portion of a larger section of tagged data.
- ID SR 11.** Ability to scale individual pieces of data with a specific number of zeros for required reporting of data (e.g. Millions, Hundreds of Thousands etc) in an instance document. For example, select the Balance Sheet and specify that when the instance document is created, the numbers that were entered into the Balance Sheet should have three zeros added to them. This way, if a financial statement says all values are in millions, all the extra zeros do not have to be typed into the tool.
- ID SR 12.** Timed backups.
- ID SR 13.** Indication that an element is a dimension-element and ability to automatically create mappings of data to or copy data to other elements.
- ID SR 14.** Should maintain and expose a user library of Units of Measure, Languages, Scenarios, Identifiers (Entity, Segments) for consistent entry in an organization.
- ID SR 15.** Ability to view the instance document in an XML View. Also, should be able to edit the XBRL directly in this view.
- ID SR 16.** Ability to render a presentation of a completed instance document.
- ID SR 17.** Internal tools for maintaining segments and reporting hierarchies.
- ID SR 18.** Useful error handling on saving and on loading an instance document. Trap non-fatal errors and allow the user to correct them in a simple manner.
- ID SR 19.** Ability to validate dimension-element ("Same-As") links and make sure the values match in the instance document.
- ID SR 20.** Promote correct entry of data through guidance from data types, context hints (UoM).
- ID SR 21.** Integration between taxonomy tool, mapping tool, instance document creation tool, and a scripting language to automate tasks.
- ID SR 22.** Ability to choose elements to validate using calculation links. A user might want to check that values in a financial statement foot. It would be conceivable that all the financial facts are not provided using the defined calculation links, so a user would have to choose which items should be summing up.

## Optional Requirements

The “Optional Requirements” contains extra features identified by the XBRL Implementation Task Force that might be useful in certain cases; however, they are not necessary for all instance document creation software.

- ID AR 1.** Automated creation of Numeric and NonNumeric contexts from type of information entered.
- ID AR 2.** Wizards to assist in identifying taxonomies, creating contexts, and entering data.
- ID AR 3.** All on screen labels external to the programming so they can easily be changed to other languages. (Globalization)
- ID AR 4.** Ability to “roll forward” a prior period or year instance, where current period entries become prior period entries and blank spaces await current period entries.
- ID AR 5.** Library functionality, security and audit trails of creation and changes.
- ID AR 6.** Audit tools that check Dimension-Elements/Same-As for discrepancies; that check sub-totals against the totals of children, and other tests; that check stock-flow links and checks the change in stock items and sees if it matches the content of the flow item, etc.
- ID AR 7.** Instant feedback mechanism for high level entry process; what has been entered/mapped, and what hasn't? Do Assets=Liabilities + Equity? Do Debits=Credits (and should they?)
- ID AR 8.** Integration with repository for storage, workflow, protection.
- ID AR 9.** Web services enabled access, query, indexing.
- ID AR 10.** Ability to run as an add-in to XML Spy so creators of other XML documents can include XBRL instance documents as data islands in their documents.
- ID AR 11.** Collaborative, group creation process tools.

## Additional Findings

### Description of Process Improvement Tool(s)

As the task force used the different types of tools, the consensus among the users was that an ideal tool should be a combination of the different types. An example is an instance document creator that allows a user to add new elements to taxonomies and create extension taxonomies. Another example is an instance document creator that is also a rendering tool. The experiences with the software also lead to the idea of having an automated instance document creation tool.

The requirements in this section are meant to be a general guideline for process improvement using tools and combining features from different types of software. The type of instance document creator being referred to is the same as in the instance document creator section, a stand alone software package.

## Instance Document Creator/Taxonomy Editor

This software would combine a full function instance document creator along with a taxonomy editor. The result would be a vast improvement for the creation process. This combination would allow a user to create both outputs in the same tool resulting in large amounts of time being saved. The base requirements below should be used in conjunction with the detail requirements listed in the respective sections for instance document creators and taxonomy editors.

1. Completely functional instance document creator (See section on instance document creator requirements.)
2. A full or lite version of a taxonomy editor. This should be a simple and intuitive add-in to the instance document creation tool. (i.e. Easier to use and should appear as a natural extension of the main user interface.)
3. Able to view the taxonomy by any of the three hierarchical linkbases (Presentation, Calculation or Definition).
4. Ability to add or create extension elements (items, abstract and tuples).
5. Ability to add or change presentation and definition linkbase orders.
6. Ability to add or change the calculation weights and parent-child relationships.
7. Ability to prohibit links.
8. Ability to add overriding labels.
9. Ability to add reference links.
10. Ability to create and validate Same As links.
11. Ability to choose XBRL taxonomies to import.
12. Validation of XBRL instance document, XML Schema and linkbases.
13. Useful and comprehensive reporting.

## Automatic Instance Document Creator/Editor

A software product that is able to load in a formatted document (PDF, RTF, etc) and automatically tag the data using a selected taxonomy could dramatically reduce the amount of time needed during the tagging process. The automatic processing would have to be highly accurate and also allow the user to easily review and overrule the suggested tagging. The simple requirements for the artificial intelligence instance document creator are listed below. These requirements are meant to be a layer on top of the instance document creator requirements and ultimately build upon the instance document creator / taxonomy editor proposed requirements.

1. Able to load a formatted document and display in a similar way. For example load a PDF document and display in a manner similar to the layout of the PDF.
2. Automatically process all of the document or sections of the document and assign tags to the data based on a database of synonyms.
3. Easily review the tags assigned to each piece of data and change the tag. Could also have the ability to tag entire rows with the same element.
4. Generate a list of close matches for each piece of data that a user could choose from during their review.
5. Manually tag any piece of data that the software was unable to match to any elements. This should allow the user to tag anything in the rendered document.

6. Automatically create contexts based on the data type and any date information available in the document.
7. Easily review the contexts with associated data the contexts are associated with. Should have the ability to easily change each instance of the context or select an entire column and choose the context as well as edit the context properties.
8. Ability to select sections of the document and specify scale factor. (The number of zeros the software should add when the instance document is created.)
9. (Nice to have) Automatically create an extension presentation linkbase using the order and hierarchy of the document that was imported and tagged.
10. (Nice to have) Automatically create label overrides to match the labels used in the imported document.
11. (Nice to have) Automatically read calculation linkbase and do the following:

## Conclusion and Wrap Up

The XBRL Implementation Task Force project was a great opportunity to exercise many different XBRL software tools. The findings concluded that there are many XBRL tools available that are able to create XBRL taxonomies and instance documents, however; there is much room for improvements and new innovations to the tools. Hopefully, the experience gained and the feedback given from this project will both inspire growth in existing tools and assist new tool creation.

