

# Applying XBRL in an Accounting Information System Design Using the REA Approach: An Instructional Case\*

JACOB PENG, *University of Michigan – Flint*

C. JANIE CHANG, *San Diego State University*

## ABSTRACT

The Church in Somewhere (CIS) is a small community church which uses an Excel spreadsheet to keep its financial records. The church administrator is considering moving from a spreadsheet accounting system to a relational database system that can easily be expanded to include more information in the future. In this paper we examine the transforming process in this hypothetical case by following a resource-event-agent (REA) modeling paradigm to create a database. We then link the REA model to financial reporting using Microsoft Access. In addition, using the financial report in the database, students prepare and validate an eXtensible Business Reporting Language (XBRL) document for CIS. Instead of applying the complex U.S. Generally Accepted Accounting Principles (GAAP) Taxonomies, Release 2009, the case uses a dedicated CIS Taxonomy to complete the mapping and tagging processes.

**Keywords** Instance document; REA modeling; Relational database; XBRL

## L'APPLICATION DU XBRL DANS LA CONCEPTION D'UN SYSTÈME D'INFORMATION COMPTABLE SELON LE MODÈLE RESSOURCE-ÉVÉNEMENT-AGENT: CAS DIDACTIQUE

### RÉSUMÉ

Church in Somewhere (CIS) est une petite église communautaire qui utilise un tableur Excel pour tenir ses registres financiers. L'administrateur de l'église songe à passer du système comptable du tableur à un système de base de données relationnelles susceptible d'être facilement élargi de manière à recevoir ultérieurement davantage d'informations. Dans ce cas hypothétique, les auteurs examinent le processus de « conversion », en suivant le paradigme du modèle ressource-événement-agent (REA), menant à la création d'une base de données. Ils relient ensuite le modèle REA à l'information financière par le truchement de Microsoft Access. En se servant du rapport financier de la base de données, ils fournissent en

---

\* We thank Professor J. Efrim Boritz, two anonymous reviewers, and participants at the 2009 Mid-Year Conference of Strategic and Emerging Technologies Section of the American Accounting Association for their helpful comments.

outre aux étudiants l'occasion de préparer et de valider un document XBRL pour CIS. Plutôt que d'appliquer les taxonomies complexes des PCGR des États-Unis, édition 2009, les auteurs utilisent dans leur cas une taxonomie propre à CIS pour réaliser les processus de cartographie et de codage.

**Mots clés :** base de données relationnelles, document d'instance, modélisation REA, XBRL

Resource-event-agent (REA) data modeling is a technique designed to build accounting information systems (AIS) in a shared data environment (McCarthy, 1982). Without using the debit-credit accounting framework, the REA modeling captures a wider range of data according to “events” that take place in everyday business transactions. Traditionally, accountants capture information only when elements in the financial statements are affected. The result of this narrow view of enterprise information is many stand-alone systems with data redundancy. The design of modern enterprise resource planning systems is an effort to reduce unnecessary redundancy in organizations. It is an example of a sophisticated version of REA-based AIS (Hurt, 2008). For this reason, most current AIS textbooks address REA modeling in addition to the traditional emphasis on business processes management.

When AIS faculty covers topics related to REA modeling, three steps are usually followed to introduce REA to students: (1) identifying business entities (i.e., resources, events, and agents); (2) determining cardinalities; and (3) implementing REA models using a relational database. Students usually are provided with business narratives in the beginning and are asked to analyze the scenario to determine the REA model.

One question often arises when students study REA data modeling: “How does the REA model help accountants prepare financial statements?” After all, an accounting information system should be useful for accountants in doing their jobs—preparing periodic financial statements and other management reports. Unfortunately, this purpose is usually skipped in many REA cases or practices. The REA model helps design a framework to collect and to store data, but it is equally important for students to be equipped with the ability to use this data—that is, to prepare reports by retrieving data from a relational database that is familiar to financial information users.

The eXtensible Business Reporting Language (XBRL) was developed for electronic communication of business data. XBRL has gradually become the de facto worldwide standard in preparing and distributing financial information to information users. The American Institute of Certified Public Accountants has consistently ranked XBRL as one of its “top ten technologies” for accounting/auditing professionals. In January 2009, the U.S. Securities and Exchange Commission (SEC) issued its final rules to require that by 2011 public companies must file XBRL-based financial statements (SEC, 2009).

Accounting professors may cover XBRL in teaching AIS courses at various levels, but many agree that a hands-on approach is the best way to learn a new technology such as XBRL. This teaching case provides guidance on how to introduce XBRL to students after an REA model is implemented. With the use of XBRL, this case provides a link between the behind-the-scenes relational databases and financial statements that is critical to most accounting and finance students. In addition, students usually are not exposed to modeling nonprofit organizations, such as hospitals, charities, and religious groups. This case provides a scenario for students to understand the simple business processes of a nonprofit organization.

### CASE SCENARIO — PART ONE

The Church in Somewhere (CIS) is a small, community-based, local church located in the Township of Somewhere, Michigan. Unlike other more established, denominational churches, CIS is not officially affiliated with other churches. Instead, CIS has a “fellowship” with other church groups who share the same vision. Regular attendance at weekly church services is about 100 people (including those who come and go). Because of the caring atmosphere and pure focus on biblical teaching, the church has experienced steady growth (in terms of the number of church members) over the past few years.

Because the church is small, it is run by a group of “deacons”. “Deacon” is not an official title; it simply refers to members who are willing to serve. Because deacons are volunteers, they do not get paid. Two years ago, one church member decided to give up his “earthly pursuit” (resigned from his job as a senior manager of an auto company) and became a full-timer to serve the church. The church deacons decided to pay him a basic subsistence of \$1,000 and a housing subsidy of \$650 per month.

Because CIS is family oriented, many activities take place in homes. That reduces the need for the church to build its own meeting hall and saves on building maintenance. However, CIS needs a meeting place to accommodate 100 people on Sundays in order for all members to worship together. Currently, CIS rents the cafeteria of a local high school to hold Sunday meetings at the cost of \$500 per month.

Similar to other religious groups, the financial needs of CIS are met by its members’ weekly donations. Mr. Steven Saiwoo, the deacon in charge of the church’s finances, uses an Excel spreadsheet to record and to manage all income and expenditures for CIS. Mr. Saiwoo is a design engineer working at a local company. He does not have much knowledge of accounting except basic bookkeeping about debits and credits. He tries his best to balance the books of CIS. Although Mr. Saiwoo is not an expert in accounting, he has a sense of responsibility and is serious about his service at CIS. Most of the time, he is quite clear about the financial situation of CIS and knows whether CIS is in deficit or not. The accounting system designed by Mr. Saiwoo is a cash-based system, because CIS does not have accruals. Exhibit 1 displays an example of the accounting system designed and used by Mr. Saiwoo.

**EXHIBIT 1**  
The Current Accounting System at CIS

A	B	C	D	E	F	G	H	I	J	K	L
Month	Week	Date	Type	Account #	Amount	Flow	Purpose	Revenue-subtotal	Expense - subtotal	Month End	Memo
1	0	1/1/2006	Deposit	Checking account -1	1,036.00	In	General				Carry-over from 2005
2	1	1/1/2006	Deposit	Checking account -2	560.00	In	Full-time Fund				Carry-over from 2005
3	1	1/1/2006	Deposit	Checking account -3	5,000.00	In	Administration				Carry-over from 2005
4	1	1/1/2006	Deposit	333094290	2,000.00	In	General	2,000.00			
5	1	1/8/2006	Deposit	cash	11.00	In	General				
6	1	2/1/15/2006	Deposit	579244208	300.00	In	Children				
7	1	2/1/15/2006	Deposit	285000641110	50.00	In	General				
8	1	2/1/15/2006	Deposit	215001962194	500.00	In	Building				
9	1	2/1/15/2006	Deposit	215002066730	200.00	In	Needy Saints				to Michael Lee
10	1	2/1/15/2006	Deposit	5402610561	1,550.00	In	Administration				
11	1	2/1/15/2006	Deposit	Children ministry	(632.00)	Ex	Children				
12	1	2/1/15/2006	546	Book purchase	(109.50)	Ex	Books				
13	1	2/1/15/2006	547	Internet fee	(9.95)	Ex	Administration	2,611.00	(751.45)		
14	1	2/1/15/2006	Transfer	Michael Lee	(200.00)	Ex	Needy Saints				
15	1	3/1/22/2006	548	one check	15.00	In	Books				
16	1	3/1/22/2006	Deposit	3270060039	50.00	In	General				
17	1	3/1/22/2006	Deposit	3270060039	50.00	In	Full-time Fund				
18	1	3/1/22/2006	Deposit	Advertising	(260.00)	Ex	Administration				
19	1	3/1/22/2006	Transfer	Insurance	(110.50)	Ex	Administration				
20	1	3/1/22/2006	548	Children ministry	(230.00)	Ex	Children	115.00	(430.00)	3544.55	
21	1	3/1/22/2006	445	cash	214.00	In	Books				
22	1	4/1/29/2006	Deposit	15540389	2,500.00	In	Administration				
23	1	4/1/29/2006	Deposit	9070446968	100.00	In	Building				
24	1	4/1/29/2006	Deposit	333094290	300.00	In	General				
25	1	4/1/29/2006	Deposit	3180846107	200.00	In	Building				
26	1	4/1/29/2006	Deposit	3180846107	100.00	In	Full-time Fund				
27	1	4/1/29/2006	Deposit	445070915	1,500.00	In	General				from 12/25/2005
28	1	4/1/29/2006	Deposit	3270060039	100.00	In	General				
29	1	4/1/29/2006	Deposit	31919705	500.00	In	General				
30	1	4/1/29/2006	549	Rent	(500.00)	Ex	Administration				
31	1	4/1/29/2006	550	Full-timer housing subsidy	(650.00)	Ex	Administration				
32	1	4/1/29/2006	551	Full-timer salary	(1,000.00)	Ex	Full-time Fund	5,514.00	(2,150.00)		
33	1	4/1/29/2006									

This is the spreadsheet designed and currently used by Mr. Saiwoo to record church financial information. This spreadsheet is available for download on the CAAA website (*See Supplementary Materials*).

Every Sunday after the meeting, Mr. Saiwoo and another deacon open the offering box and record donations (cash and/or cheques) made by church members on that day. Because the Bible says “when you give alms, do not let your left hand know what your right hand is doing” (Matt 6:3), Mr. Saiwoo does not want to record who makes the donation. However, for tax purposes, CIS does need this information so that CIS can prepare receipts if requested by the member. To solve this paradox, Mr. Saiwoo records only the account number shown at the bottom of each check. By doing so, Mr. Saiwoo and other deacons can avoid knowing who made the donations. If it is a cash donation, an envelope is used for a member to indicate the dollar amount and purpose of the donation.

When members make donations, they can specify the purpose(s) of their donations. Some members would like to donate one offering for different purposes. For example, the donation can be dedicated to “children’s services”, “full-time staff support”, “church administration”, etc. Hence, Mr. Saiwoo uses several “funds” to record the indicated use of money. Several common funds used by Mr. Saiwoo are:

- Administration fund
- Books fund
- Full-time staff fund
- Building fund (for the purpose of building a meeting hall)
- Children’s ministry fund
- Needy saints fund (sometimes, members make donations for the purpose of helping other members financially)
- General fund (this fund is used if there’s no special purpose specified by donors)

Mr. Saiwoo also matches expenditures with the funds he has created. For example, he pays full-timer salary from the “full-time staff fund” and pays monthly rental from the “administration fund”. If there is a shortage in a particular fund, Mr. Saiwoo uses the “general fund” to make up the shortage. In other words, the “general fund” can be used for any legitimate purpose.

Although the aforementioned funds are the most commonly ones used in day-to-day operations, Mr. Saiwoo would like to have the flexibility to include other funds in the future. At the end of the month, Mr. Saiwoo writes cheques to pay for operational expenses and to some church members if there were offerings dedicated to financially help those members.

For the periodical preparation of financial statements, Mr. Saiwoo summarizes all income (from offerings) and expenses and calculates the net income or loss for that particular period. Again, because the church has no accruals, only cash income and expenses are recorded and reported. Exhibit 2 shows an example of such a financial report.

**EXHIBIT 2**

## The Current Financial Statement

<b>The Church in Somewhere</b>		
Somewhere, Michigan		
<b>Monthly Statement of Income and Expenses</b>		
For the month ended January 31, 2006		
<b>Income</b>		
Offerings (Administration)	\$	4,050.00
Offerings (Books)		229.00
Offerings (Building fund)		800.00
Offerings (Children's ministry)		300.00
Offerings (Full-timer)		150.00
Offerings (General)		4,511.00
Offerings (Needy saints)		200.00
<b>Total Income</b>	<b>\$</b>	<b>10,240.00</b>
<b>Expenses</b>		
<u>Operations—administration</u>		
Advertising	\$	260.00
Insurance		110.50
Internet		9.95
Rent		500.00
<u>Operations—books</u>		
Book purchase		109.50
<u>Operations—children's ministry</u>		
Children's supplies		862.00
Total		1,851.95
<u>Needy saints</u>		
Needy saints	\$	200.00
Total		200.00
<u>Full-timer</u>		
Housing subsidy	\$	650.00
Full-timer salary		1,000.00
Total		1,650.00
<b>Total Expenses</b>		<b>3,701.95</b>
<b>NET INCOME</b>	<b>\$</b>	<b>6,538.05</b>

This is an example of a monthly financial statement prepared by Mr. Saiwoo based on the spreadsheet he uses.

**Requirements**

- 1) Study the spreadsheet accounting system currently used by Mr. Saiwoo (Exhibit 1, *See Supplementary Materials*). Identify resources and agents for two events: cash donations and cash expenses (learning objective 1).

- 2) Create an REA model for the accounting system of CIS (learning objective 1).
- 3) Implement the REA model from item (2) above using a relational database (learning objective 2). Enter data from the spreadsheet to the database you have created.
- 4) Prepare a query that summarizes all cash donations and expenses by purpose (fund) (learning objective 2).

### **CASE SCENARIO—PART TWO**

At the end of each month, one thing that troubles Mr. Saiwoo when he prepares periodic financial statements is that he has to enter the same data multiple times. For example, he uses an Excel function to summarize all monthly offerings, writes the numbers down; then proceeds to the next month, writes a second number down. Eventually he gets three monthly numbers for a quarter, and then he calculates the quarterly total for donations. For expenses, he does the same thing manually. Once he has all numbers on hand, he reconciles them with bank statements. If there are any discrepancies, he repeats the processes until he resolves the problem. When all numbers eventually agree, Mr. Saiwoo then enters all numbers to their appropriate places in the financial statement. Sometimes he doesn't notice an error until the statement is ready. Errors that occur in transferring data from one place to another easily take several hours for Mr. Saiwoo to trace and correct.

XBRL is a solution for Mr. Saiwoo's problem. Tagging financial information once allows the tagged information to be used multiple times without ever re-entering the information. In this part of the case, you will prepare an XBRL statement for CIS. After you finish this hands-on portion of the case, you should have an appreciation for the role XBRL plays in the financial reporting community.

To start, you need to understand several components of XBRL:

1. \*.xml file = this is an XBRL statement (called an "instance document") you will create in this project. It contains all "tagged" financial information.
2. \*.xsd file = this is XBRL's taxonomy file. The taxonomy is like a dictionary of financial reporting elements (such as names of accounts used by accountants). Taxonomy defines these elements and their relationships. A complete taxonomy includes XML schema and link bases. You will download an XBRL taxonomy designed specifically for CIS from the case website (see Supplementary Materials).
3. \*.xsl file = this is the style sheet used to present the instance document in a human-readable form. Because the style sheet is context free, it can be applied to multiple instance documents.

Preparing an XBRL-based report (i.e., instance document) requires not only technical know-how but also professional judgment. Recall that XBRL is designed

to assist accountants in preparing financial reports efficiently for electronic transmission. An instance document is a collection of financial data typically seen in financial statements. The process starts with analyzing what financial information should be included in an instance document. Typically, those numbers appear in financial statements and information disclosed in footnotes.

Briefly speaking, taxonomy selection, mapping, tagging, and validation are four important steps that need to be followed to prepare an XBRL-based financial report (Mascha, Janvrin, Plouff, and Kruger, 2009). Organizations that are required to file XBRL reports to authorities need to start by identifying the financial information to tag. After identifying which financial information to tag, preparers choose a standard taxonomy (such as U.S. GAAP Taxonomies, Release 2009) and load it into the XBRL software. After setting up contexts of the company (i.e., entity name, financial reporting period, currency), each account value is separately mapped to a specific XBRL element from the standard taxonomy. Tagging is then applied to use appropriate tags to represent financial data. If the standard taxonomy does not include an appropriate element, the preparers may choose to extend the standard taxonomy by creating a new XBRL element. Validation is done at the end to make sure that all required tags are applied and XBRL specifications are satisfied. Detailed XBRL preparation steps and how this case fits into the illustrated steps are presented in Exhibit 3.

As illustrated in Exhibit 3, to tag financial statements companies/organizations usually download the taxonomies from [xbrl.org](http://www.xbrl.org), the nonprofit consortium that develops taxonomies for each industry in each country. Because taxonomies are “eXtensible”, any organization can extend the existing taxonomy to meet organization-specific needs.

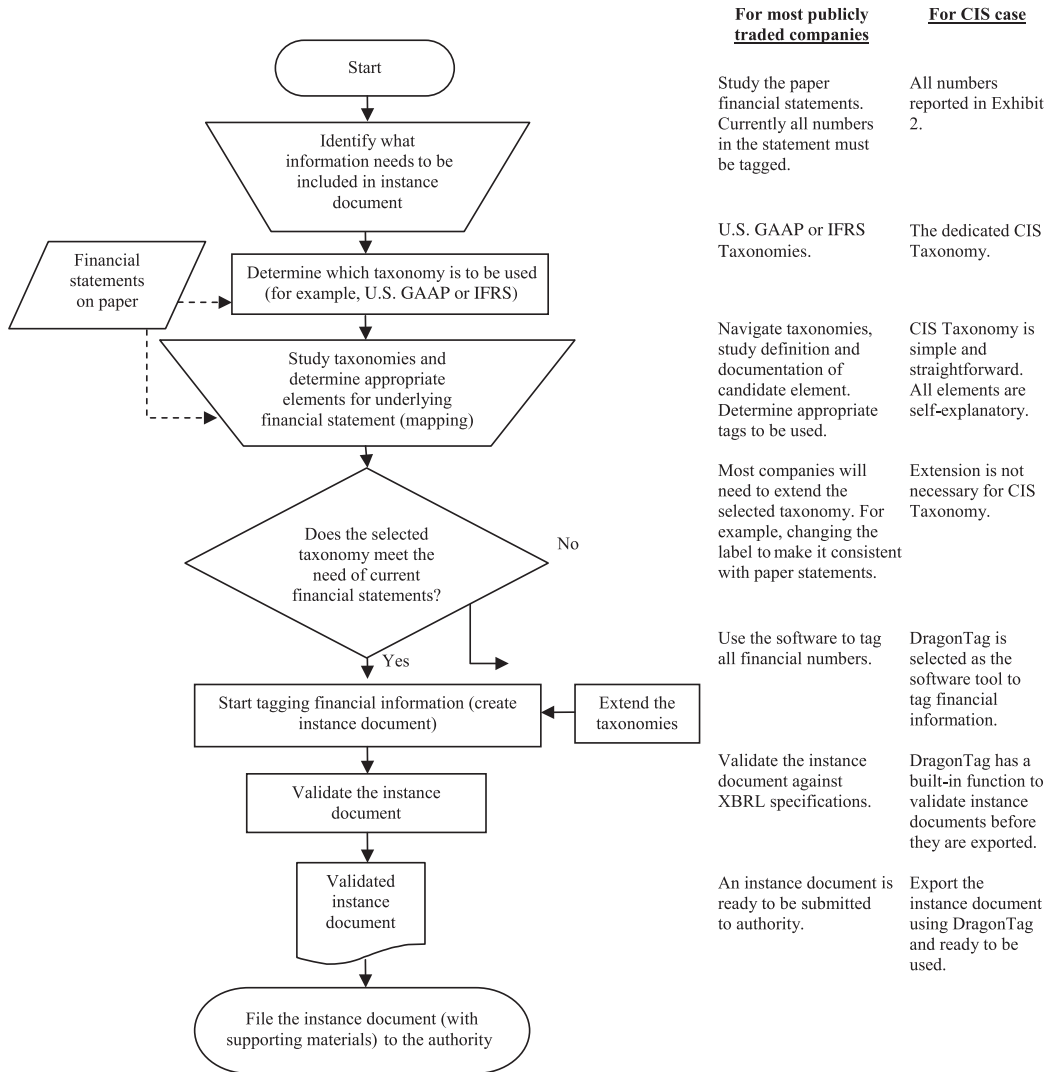
For the purpose of this case, you will download a special taxonomy prepared by the case author. Compared with the current U.S. GAAP Taxonomies, Release 2009 (available at <http://www.xbrl.us/taxonomies/Pages/US-GAAP2009.aspx>), the CIS taxonomy is much smaller and more manageable, with the sole purpose of tagging CIS financial statements. Although the CIS taxonomy is simple, it follows the XBRL standards applicable when it was created. Exhibit 4 shows the structure of CIS taxonomy reporting elements. Download the zipped taxonomy file to your computer (*see Supplementary Materials*), and make sure that you unzip all files to the same location in your computer.

To prepare a quarterly statement for CIS, the first thing you need to do is to retrieve data from your relational database. Structured query language is such a tool designed as an interface between you and the relational database. Refer to the example statement in Exhibit 2; CIS reports its financial operations by fund. As a result, you need to create queries summarizing income and expenses by different funds. Examples of the income and expense query results are in Exhibit 5.

Although Microsoft Office supports XML, you need to use special software to prepare XBRL instance documents in order to meet XBRL specifications.



**EXHIBIT 3**  
XBRL Document Preparation Process

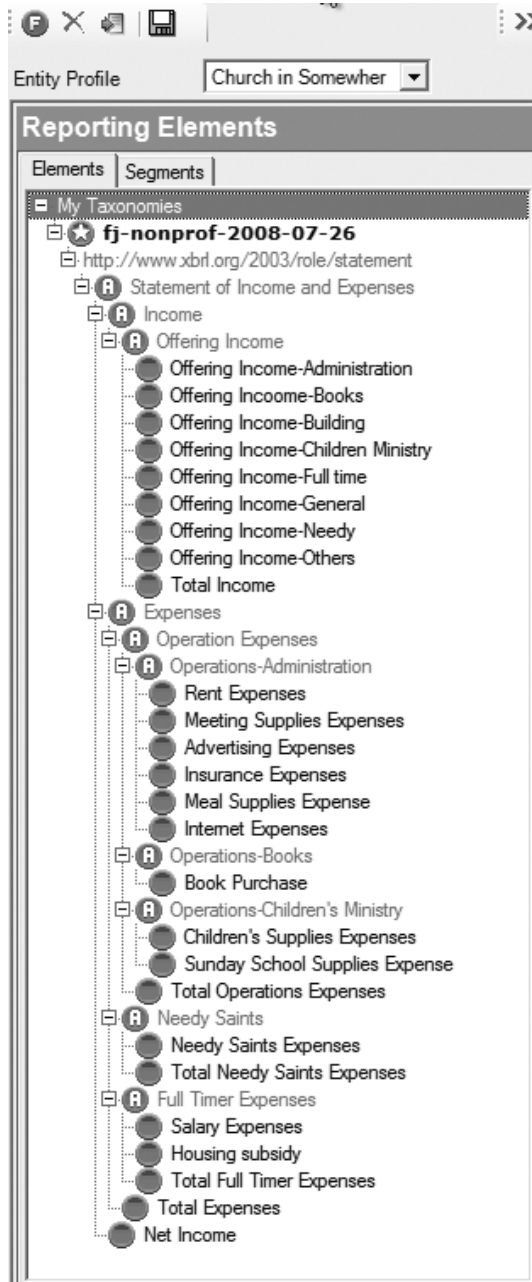


DragonTag by Rivet Software is one such available tool. DragonTag offers a full-version, 30-day trial at no cost to individuals.<sup>1</sup> DragonTag is selected for this case because it is currently the most common tool used by SEC’s voluntary filing program participants (Phillips, Bahmanziari, and Colvard, 2008). Because DragonTag

1. It can be downloaded at the following address: <http://www.rivetsoftware.com/Products/downloads.aspx#Dragon%20Tag>

**EXHIBIT 4**

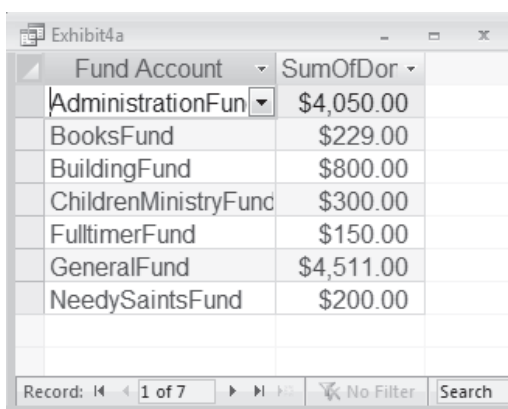
## CIS Taxonomy Reporting Elements



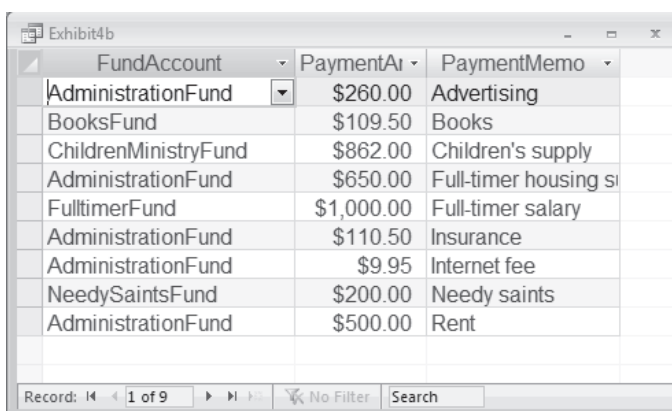
functions as a Microsoft Excel and Word plug-in, you will need to export your data (income and/or expense query) from the database to a spreadsheet. Appendix 1 provides instructions on how to export query results to an Excel spreadsheet.

**EXHIBIT 5**

## Income Summary by Funds Query Result

**Panel A:** Income query


Fund Account	SumOfDor
AdministrationFun	\$4,050.00
BooksFund	\$229.00
BuildingFund	\$800.00
ChildrenMinistryFund	\$300.00
FulltimerFund	\$150.00
GeneralFund	\$4,511.00
NeedySaintsFund	\$200.00

**Panel B:** Expenses query


FundAccount	PaymentAt	PaymentMemo
AdministrationFund	\$260.00	Advertising
BooksFund	\$109.50	Books
ChildrenMinistryFund	\$862.00	Children's supply
AdministrationFund	\$650.00	Full-timer housing s
FulltimerFund	\$1,000.00	Full-timer salary
AdministrationFund	\$110.50	Insurance
AdministrationFund	\$9.95	Internet fee
NeedySaintsFund	\$200.00	Needy saints
AdministrationFund	\$500.00	Rent

**Requirements**

- 5) Obtain the taxonomy designed for the CIS and an instance document creation tool (learning objective 3).
- 6) Tag CIS financial statements with CIS taxonomy (learning objective 3).
- 7) Validate the CIS XBRL report against the CIS taxonomy (learning objective 3).

For the purpose of this case, download the CIS Taxonomy files (*see Supplementary Materials*) and save them to your hard drive. Then, in DragonTag, start the program by selecting “Start Excel with DragonTag Add-In”. Then open the Excel file

with income and expense data you exported from the database. Follow the instructions provided in Appendix 2 to prepare and to validate a CIS XBRL instance document for January 2006.

## CASE LEARNING OBJECTIVES AND IMPLEMENTATION GUIDANCE

### Case Overview and Teaching Objectives

Designing an AIS using the REA approach has several advantages. It helps students to think outside of the box in regard to sharing information in a database environment with which accountants are traditionally not familiar. The REA modeling developed by McCarthy in 1982 addresses the need for modeling transaction cycles in a database environment which produces data to support decision making in various business functions.

There are different ways to teach REA modeling. Some instructors mention REA in their AIS course briefly, while others devote almost an entire course to this topic. Additionally, instructors use different business scripts to enhance students' learning of modeling techniques. The majority of these business scripts or scenarios are in the realm of transaction cycles in traditional accounting settings, namely sales/revenue, purchase/payment, human resources, and financing cycles. Students seldom are exposed to modeling other types of accounting in addition to sale-purchase scenarios. This case provides an REA practice set from a different perspective — in a nonprofit setting. In addition to the most common approach, using business scenarios for teaching REA, this case provides an Excel spreadsheet with which most students are familiar. This helps students to start thinking about resources, events, and agents involved in the business processes and then to transform the spreadsheet-based accounting into a database based on an REA model.

After students build their database accounting system given the REA modeling practice set in Part I of the case, Part II of this case requires students to implement an accounting system applying an emerging technology: XBRL.

In most cases, accountants might not need to have the technical know-how to design an accounting system from scratch; nevertheless, they definitely need to have the ability to interact with the system — that is, the ability to retrieve data from the system and to organize that data into useful information. Part II of the case helps students practice their query language skills to retrieve data from the database designed in Part I and then to use the data to apply XBRL taxonomy to create an XBRL instance document in preparing financial statements.

Many XBRL cases available for educational cases use Commercial and Industrial (CI) industry entry point<sup>2</sup> of U.S. GAAP Taxonomies, Release 2009 (the most

---

2. An industry entry point in the taxonomy is an XBRL file that brings together a set of files to define relationships among elements. This type of file has the extension “.xsd”. Currently there are five different entry points as defined in US GAAP Taxonomies, Release 2009: banking and savings institutions, brokers and dealers, commercial and industrial, insurance, and real estate.

current release of the U.S. GAAP as of this writing) for creating instance documents. Although using U.S. GAAP CI Taxonomy is realistic and it is the same taxonomy used by companies to submit their XBRL-based financial reports, the complexity of CI taxonomy often creates hurdles for students the first time they are exposed to XBRL. This case follows the same approach by requiring students to create an XBRL instance document, but students use a much simpler taxonomy prepared by the case authors. This approach provides students with a basic understanding of how XBRL works and assists them in exploring XBRL instance documents and taxonomy. More importantly, students (as financial statement preparers) can acquire hands-on experience in creating an XBRL instance document without getting lost in the complex CI taxonomy.

The learning objectives of this case are: (1) to transform a spreadsheet-based accounting system to an event-based AIS, (2) to implement the REA model for a nonprofit organization, and (3) to tag, validate and export financial data for XBRL-based financial reporting (i.e., to create an XBRL instance document).

### **Implementation Guidance**

Students are asked to assume the role of a consultant who is responsible for proposing a new AIS for the client, CIS. Students first try to understand the background of the organization based on the case scenario in Part I. Then students need to study the organization's current information system using the Excel spreadsheet provided. The analyses and discussions in the Instructor Teaching Notes provide more detailed information about each requirement in the case. While this case is designed to be completed by individual students, small groups can also be used to implement this case. Skills required in completing the case are discussed in the Instructor Teaching Notes.

### **CLASSROOM TESTING**

A survey was conducted to examine the learning effectiveness of using this case on students' understandings of REA modeling and of XBRL. Undergraduate students taking an AIS course at two state universities participated in the survey. In general, students read and understood the case requirements (on a 5-point Likert-type scale with 1 as "strongly disagree" and 5 as "strongly agree"; mean = 4.04), and the taxonomy used in this case is easy to understand and easy to use (mean = 4.18). Also, we asked students to indicate (on a 5-point Likert-type scale with 1 as "strongly disagree" and 5 as "strongly agree") the extent to which they agreed or disagreed with statements regarding their experiences with the case. Overall, the students agreed that the case was interesting (mean = 3.81), good learning experience (mean = 4.05), helped understand REA modeling (mean = 4.27), and helped understand XBRL (mean = 4.18). The student survey questionnaire is presented in Exhibit 6.

**EXHIBIT 6**

## Student Survey Conducted after Completing the Case

Please provide your feedback on the case to the instructor for continuous improvement. Your experience and opinions are valuable! Please take a few minutes to answer the following questions. Your responses will be anonymous and the results will only be used to assess the effectiveness of using this case in AIS courses.

Based on your experiences while working on the Church in Somewhere (CIS) REA-XBRL case, please indicate your level of agreement with each of the following statements using the five-point scale provided (please circle one response for each item).

1. I read the case completely before I worked on the case. (1). No. I did not read any of the case. (2). No. I did not read most of the case. (3). I glanced over most of the case. (4). Yes. I glanced over the whole case. (5). Yes. I read the case completely.
2. After I had read the case, I understood the case requirements. (1). Strongly disagree. (2). Disagree. (3). Neutral. (4). Agree. (5). Strongly agree.
3. The case helped me practice REA modeling skills. (1). Strongly disagree. (2). Disagree. (3). Neutral. (4). Agree. (5). Strongly agree.
4. The case helped me understand REA modeling better. (1). Strongly disagree. (2). Disagree. (3). Neutral. (4). Agree. (5). Strongly agree.
5. The case helped me learn XBRL (eXtensible Business Reporting Language). (1). Strongly disagree. (2). Disagree. (3). Neutral. (4). Agree. (5). Strongly agree.
6. The XBRL taxonomy used in the case was easy to understand and easy to use in completing the project. (1). Strongly disagree. (2). Disagree. (3). Neutral. (4). Agree. (5). Strongly agree.
7. The case was interesting. (1). Strongly disagree. (2). Disagree. (3). Neutral. (4). Agree. (5). Strongly agree.
8. Working on the case was a good learning experience. (1). Strongly disagree. (2). Disagree. (3). Neutral. (4). Agree. (5). Strongly agree.
9. Your age: (1). < 20. (2). 20 – 23. (3). 24 – 26. (4). 27 – 30. (5). > 30.
10. Your gender: (1). Male. (2). Female.
11. Your academic classification (1). Freshman. (2). Sophomore. (3). Junior. (4). Senior. (5). Graduate. (6). Others.
12. What was your grade in the AIS course you took?
13. Did you find anything in the case unclear?
14. Any other comments?

### SUPPLEMENTARY MATERIALS

Supplementary materials (files) for this case are available for download on the CAAA website at: <http://www.caaa.ca/AccountingperspectivesCAP/CasesTNSM>.

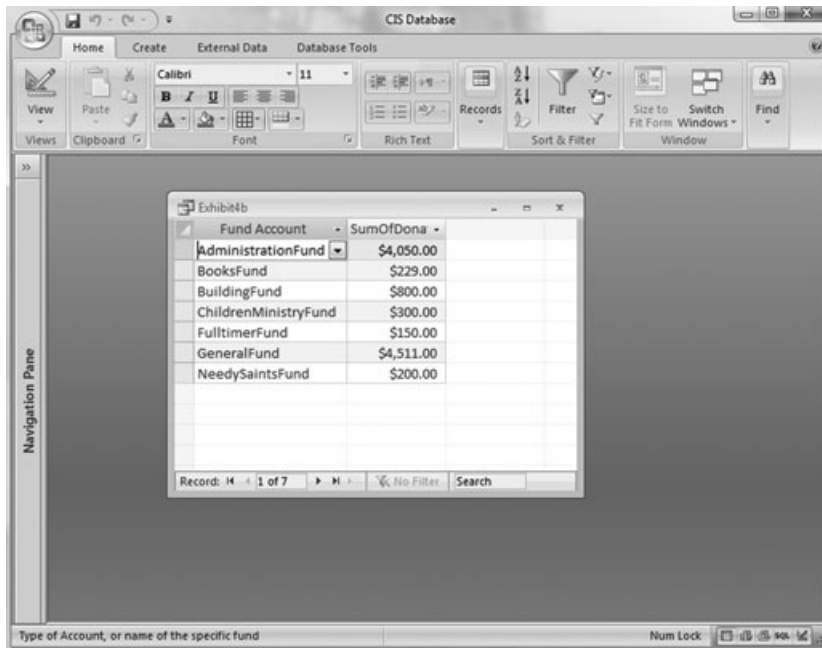
### TEACHING NOTES

Teaching Notes for instructional cases are not published in the journal but are made available to full CAAA member subscribers in a password-protected area of the CAAA website.

If you are a full member of the CAAA with a subscription to *Accounting Perspectives* and wish to obtain a copy of the Teaching Notes, please go to <https://www.caaa.ca/en/journals-and-research/accounting-perspectives-ap/teaching-notes/> to log in and access the notes.

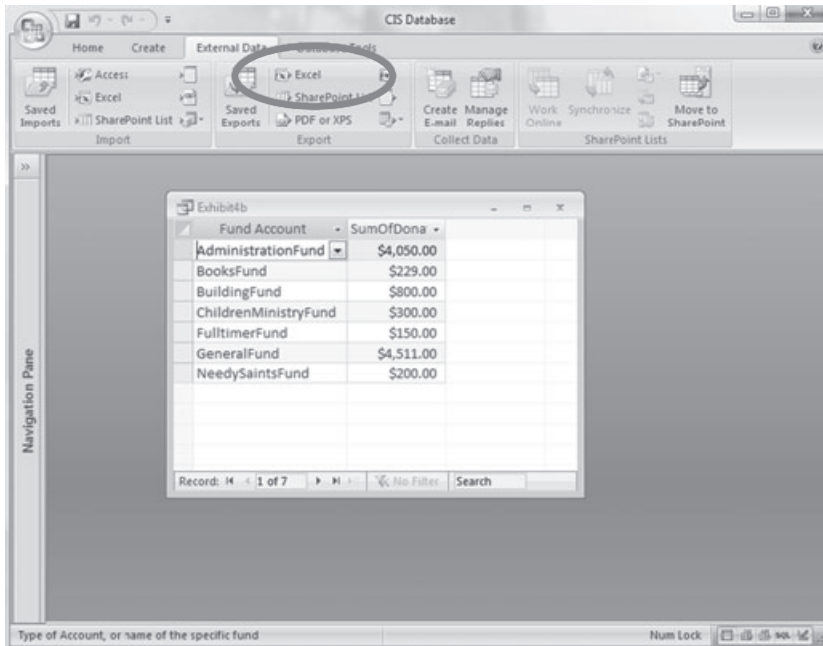
### APPENDIX 1: EXPORTING QUERY RESULTS TO EXCEL

Step 1: Run execute the query to plan to export.

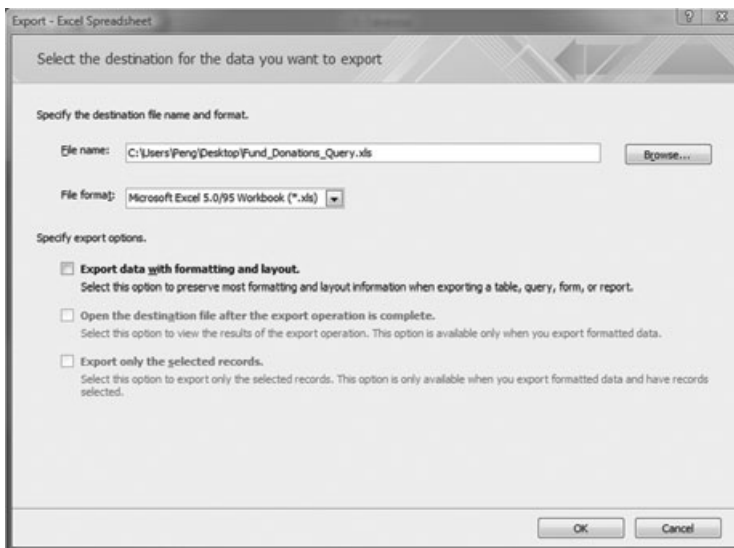


## APPENDIX 1: (Continued)

Step 2: Click “External Data” → Export to Excel



Step 3: Choose the file name and location, click “OK” to finish exporting data to Excel.



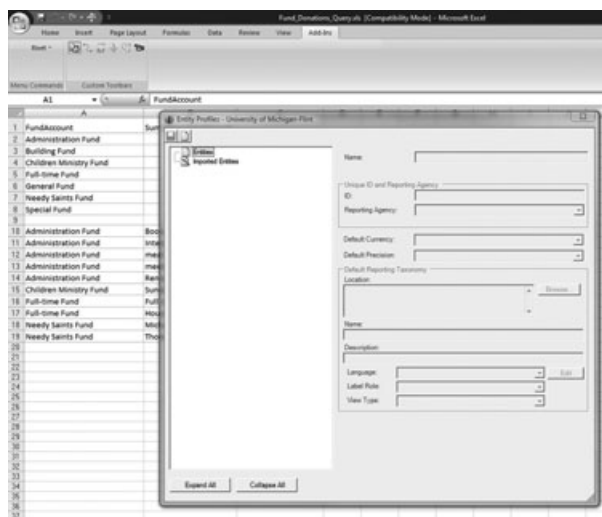


### APPENDIX 2: STEP-BY-STEP TUTORIALS

1. Start DragonTag: in Excel, open the file with income and expense data. Calculate total donations, total expenses by category, total expenses and net income using Excel's formula (as shown in the screenshot below). Click "Add-ins" to find DragonTag menus.

FundAccount	SumOfDonationAmount	PaymentAmountOfSum	PaymentMemo
AdministrationFund	4050	260	Advertising
BooksFund	329	600	Full-time housing subsidy
BuildingFund	800	138.5	Insurance
ChildrenMinistryFund	300	9.35	Internet fee
FulltimeFund	150	500	Rent
GeneralFund	4511	109.5	Books
NeedySaintsFund	200	862	Children's supply
Total Income	10240	3000	Full-time salary
		200	Needy saints
Total Operations		1851.95	
Total Needy saints		200	
Total Full timer		1850	
Total expense		1701.95	
Net income		6538.05	

2. Click the home icon to open entity profile setup page.

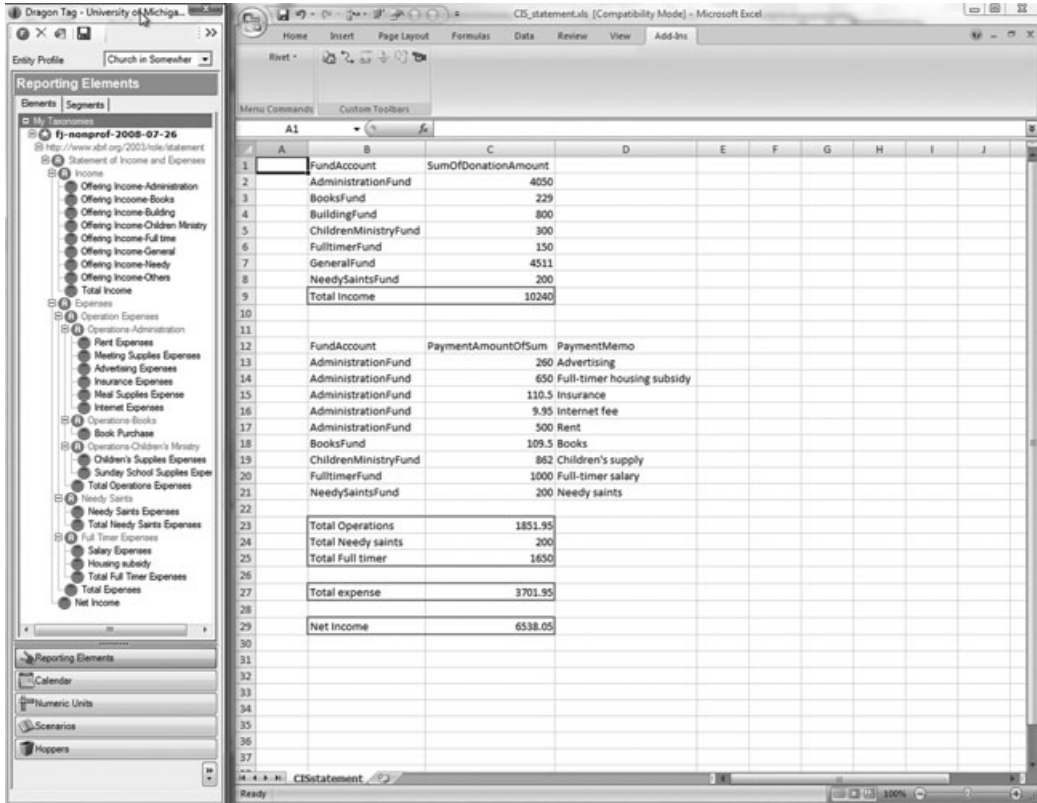


## APPENDIX 2: (Continued)

3. Click “Create New Entity” (📄 icon) to create a new entity. Enter information about the entity (Church in Somewhere) you are about to work on. Make up something in the ID and Reporting Agency (since you are not sending this XBRL report to any authority except your instructor, you can just enter <http://www.cis.org/xbml> as the reporting authority). Note that the most important step here is to determine “Default Reporting Taxonomy”. Direct the default taxonomy to your hard drive location where you saved the CIS taxonomy you downloaded. The taxonomy’s file name is “fj-nonprof-2008-07-26.xsd”. In addition, you need to select appropriate currency (USD) and precision (the number of digits as shown).

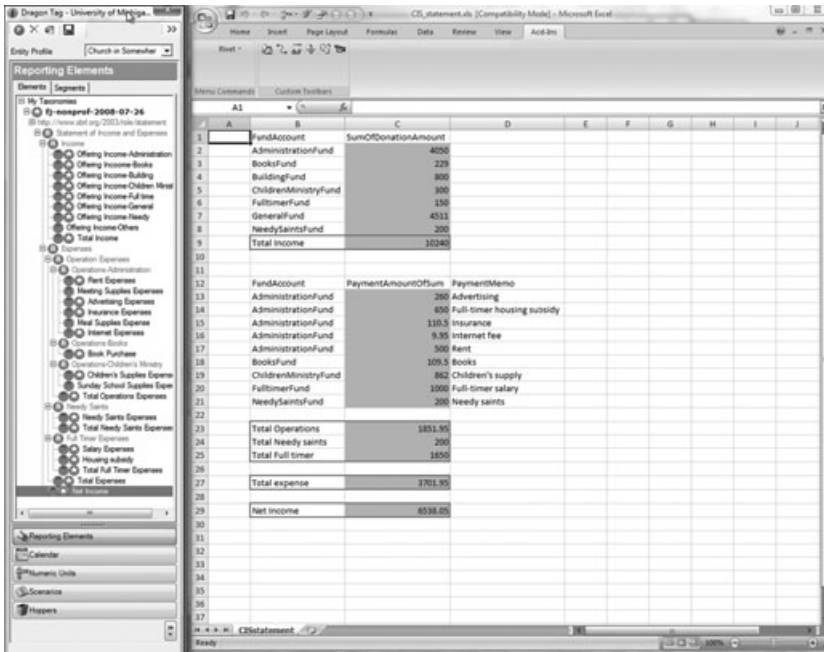
4. When DragonTag successfully locates the xsd file (CIS taxonomy), it gives a green light in “Default Reporting Taxonomy” screen. Now you are ready to “tag” financial data in this spreadsheet. Click “Markup data” button (📄 icon). The “Reporting Elements” screen shows up. If you expand all taxonomy elements, you can see that the CIS taxonomy is a simple taxonomy with only 37 elements.


APPENDIX 2: (Continued)

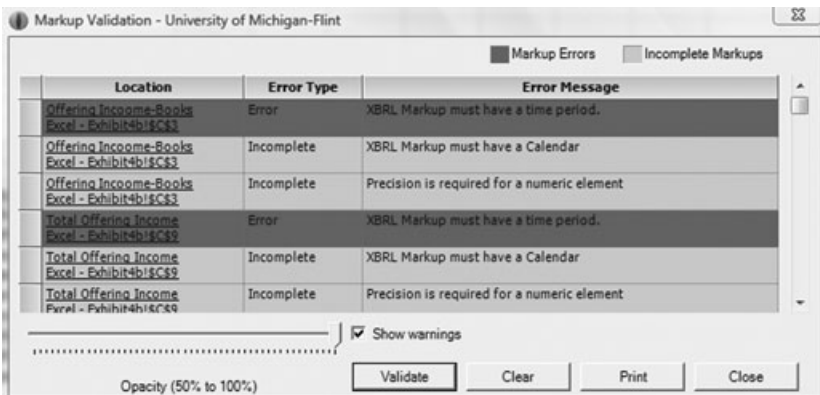


- Now, identify the correct tag from the taxonomy, then drag and drop it on an appropriate cell in your spreadsheet. Please remember that only cells containing numbers need to be tagged – XBRL is about data, not presentation. Cells that are tagged will be highlighted by DragonTag. Tagging elements used will show a star symbol in front of the element name.

APPENDIX 2: (Continued)

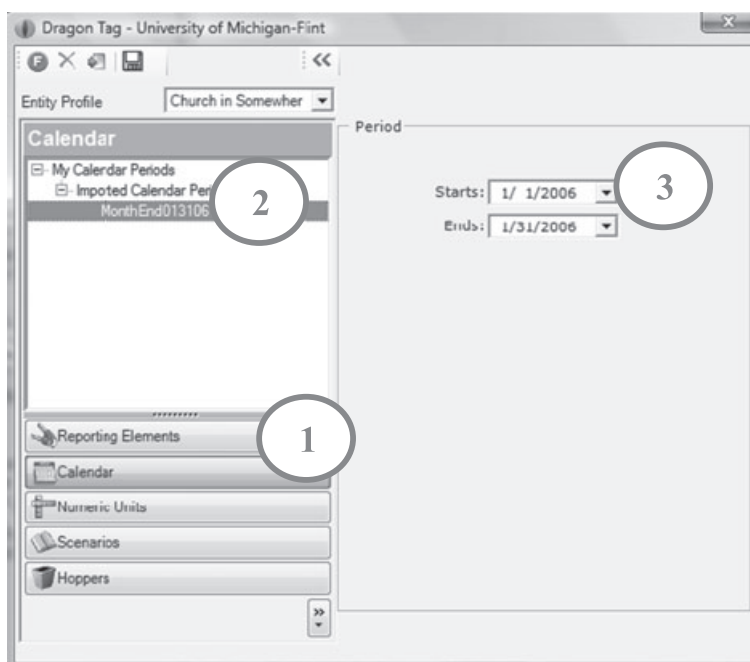


- When you finish the “mapping and tagging” (drag and tag) process, you have almost completed an XBRL instance document for CIS which reports all financial numbers from the paper-based statement in XBRL format. Before you can export the instance document, you need to “validate” the document to make sure it meets all XBRL specifications. DragonTag provides two layers of validation: validate that all necessary XBRL pieces have been applied, and all numeric data values are accurate according to the calculation rules embedded in the taxonomy. Click the  icon and press “Validate” button to start the validation process. The software will detect that some XBRL information are missing—the time period, calendar, and precision, if you did not specify those when you created the entity.



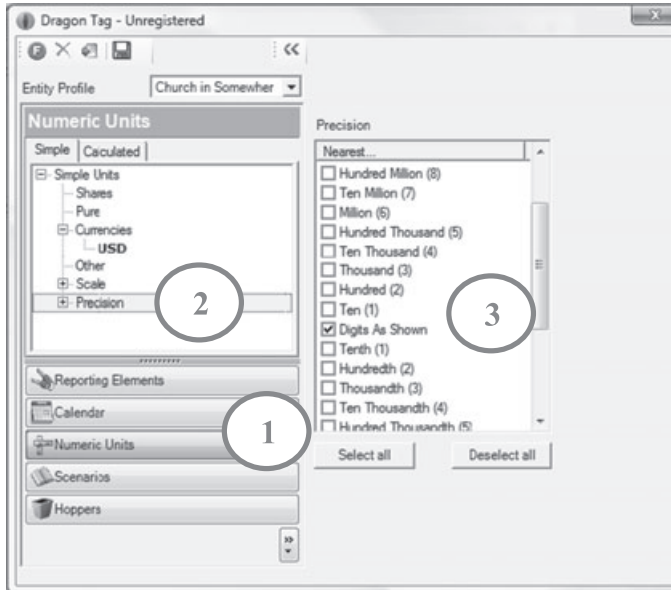
**APPENDIX 2: (Continued)**

7. The reason you may experience these problems is that a complete XBRL tag for each financial reporting element needs to have at least three pieces of context information: time period which this financial reporting element covers, currency, and precision of the number reported in addition to the financial reporting tags. To correct the problem you will need to add calendar, currency, and precision tags to each cell. In DragonTag, separate context elements can be used to tag each number. However, this can be a tedious process if done manually. DragonTag provides a feature to create “hoppers” to help complete this process. Basically, a hopper is a group of context elements that can be used to tag all data directly. It functions like a macro that speeds up the tagging process. To create a hopper to add context information to each cell, click “calendar” tag on the left, and manually create a reporting period by right clicking on “My Calendar Periods” and select “Add Time Period”. Name the time period “CIS” and specify the date range. In the CIS case, it covers 2006-01-01 to 2006-01-31.

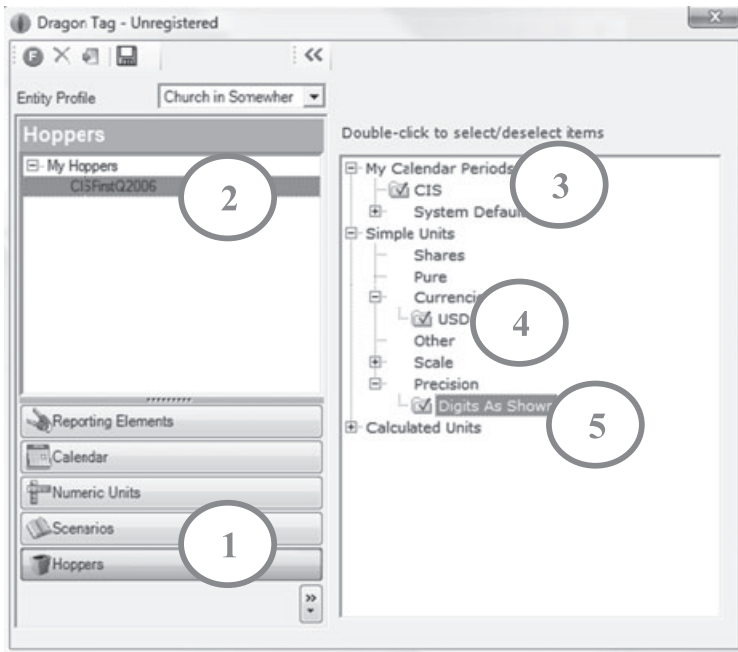


8. Create a second context element by clicking “Numeric Units” tag on DragonTag. Select “Digits as Shown” under “Precision” because that is the case for CIS.

**APPENDIX 2: (Continued)**

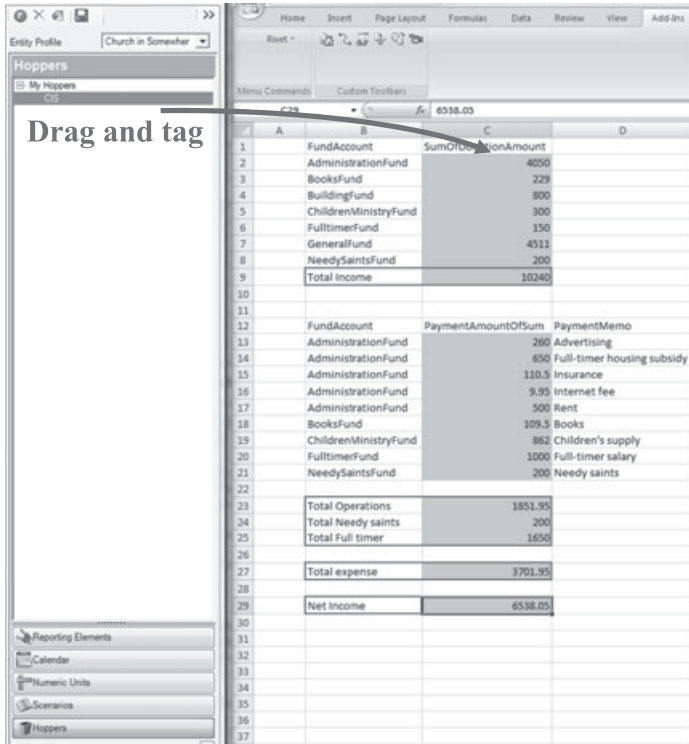



9. Finally, click “Hoppers” and create a new hopper. Give the hopper an appropriate name. Then select the context elements that need to be included in this hopper by double-clicking each context element.

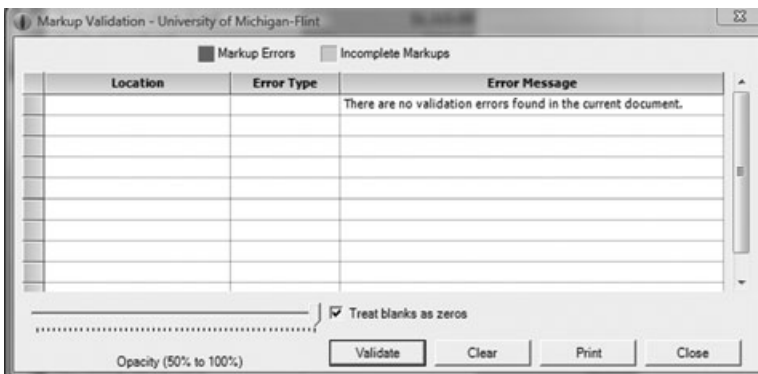


**APPENDIX 2: (Continued)**

10. A hopper is created. Instead of tagging each data cell in Excel with individual context elements, this hopper can be used to add context elements to each data cell (by dragging and dropping the hopper you've just created to every cell you previously tagged).



11. When you finish tagging all information, click “Validate Markup” button again to validate your work (  icon). If there are no validation errors, congratulations! You just finished an XBRL instance document for CIS!



12. The last step is to find the “Export XBRL” function (📄 icon), and export your XBRL instance document! If you open your XBRL instance document in a web browser, it should look like this:

```
<?xml version="1.0" encoding="utf-8" ?>
<!-- XBRL document created with Dragon Tag (R) XBRL Enabler by Rivet Software version 3.0.0.47 -->
<!-- Based on XBRL 2.1 -->
<!-- Created on: 7/31/2009 2:49:06 PM -->
- <xbrl xmlns="http://www.xbrl.org/2003/instance" xmlns:link="http://www.xbrl.org/2003/linkbase" xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:fj-nonprof="http://xbrl.jacobpeng.net" xmlns:iso4217="http://www.xbrl.org/2003/iso4217">
  <link:schemaRef xlink:type="simple" xlink:href="fj-nonprof-2008-07-26.xsd" />
  <!-- Context Section -->
  - <context id="MonthEnd013106">
    - <entity>
      <identifier scheme="http://www.cis.org/xbrl">CIS</identifier>
    </entity>
    - <period>
      <startDate>2006-01-01</startDate>
      <endDate>2006-01-31</endDate>
    </period>
  </context>
  <!-- Unit Section -->
  - <unit id="USD">
    <measure>iso4217:USD</measure>
  </unit>
  <!-- Tuple Section -->
  <!-- Element Section -->
  <fj-nonprof:AdminAdvertise contextRef="MonthEnd013106" unitRef="USD" decimals="INF">260</fj-nonprof:AdminAdvertise>
  <fj-nonprof:AdminInsurance contextRef="MonthEnd013106" unitRef="USD" decimals="INF">110.5</fj-nonprof:AdminInsurance>
  <fj-nonprof:AdminRent contextRef="MonthEnd013106" unitRef="USD" decimals="INF">500</fj-nonprof:AdminRent>
  <fj-nonprof:BookExpense contextRef="MonthEnd013106" unitRef="USD" decimals="INF">109.5</fj-nonprof:BookExpense>
  <fj-nonprof:ChildrenSupplyExpenses contextRef="MonthEnd013106" unitRef="USD" decimals="INF">862</fj-nonprof:ChildrenSupplyExpenses>
  <fj-nonprof:FTHousingSubsidy contextRef="MonthEnd013106" unitRef="USD" decimals="INF">650</fj-nonprof:FTHousingSubsidy>
  <fj-nonprof:FTSalary contextRef="MonthEnd013106" unitRef="USD" decimals="INF">1000</fj-nonprof:FTSalary>
  <fj-nonprof:InternetExpenses contextRef="MonthEnd013106" unitRef="USD" decimals="INF">9.95</fj-nonprof:InternetExpenses>
  <fj-nonprof:NeedySaintsExpenses contextRef="MonthEnd013106" unitRef="USD" decimals="INF">200</fj-nonprof:NeedySaintsExpenses>
  <fj-nonprof:NetIncome contextRef="MonthEnd013106" unitRef="USD" decimals="INF">6538.05</fj-nonprof:NetIncome>
  <fj-nonprof:Offering-admin contextRef="MonthEnd013106" unitRef="USD" decimals="INF">4050</fj-nonprof:Offering-admin>
  <fj-nonprof:Offering-books contextRef="MonthEnd013106" unitRef="USD" decimals="INF">229</fj-nonprof:Offering-books>
  <fj-nonprof:Offering-building contextRef="MonthEnd013106" unitRef="USD" decimals="INF">800</fj-nonprof:Offering-building>
  <fj-nonprof:Offering-children contextRef="MonthEnd013106" unitRef="USD" decimals="INF">300</fj-nonprof:Offering-children>
  <fj-nonprof:Offering-fulltime contextRef="MonthEnd013106" unitRef="USD" decimals="INF">150</fj-nonprof:Offering-fulltime>
  <fj-nonprof:Offering-general contextRef="MonthEnd013106" unitRef="USD" decimals="INF">4511</fj-nonprof:Offering-general>
  <fj-nonprof:Offering-Needy contextRef="MonthEnd013106" unitRef="USD" decimals="INF">200</fj-nonprof:Offering-Needy>
  <fj-nonprof:TotalExpenses contextRef="MonthEnd013106" unitRef="USD" decimals="INF">3701.95</fj-nonprof:TotalExpenses>
  <fj-nonprof:TotalFullTimerExpenses contextRef="MonthEnd013106" unitRef="USD" decimals="INF">1650</fj-nonprof:TotalFullTimerExpenses>
  <fj-nonprof:TotalNeedySaintsExpenses contextRef="MonthEnd013106" unitRef="USD" decimals="INF">200</fj-nonprof:TotalNeedySaintsExpenses>
  <fj-nonprof:TotalOfferingIncome contextRef="MonthEnd013106" unitRef="USD" decimals="INF">10240</fj-nonprof:TotalOfferingIncome>
  <fj-nonprof:TotalOperationsExpenses contextRef="MonthEnd013106" unitRef="USD" decimals="INF">1851.95</fj-nonprof:TotalOperationsExpenses>
  <!-- Footnote Section -->
</xbrl>
```

Local intranet | Protected Mode: Off 100%

## REFERENCES

- Hurt, R. L. 2008. *Accounting information systems: Basic concepts and current issues*. New York: McGraw-Hill.
- McCarthy, W. E. 1982. The REA accounting model: A generalized framework for accounting systems in a shared data environment. *Accounting Review* (July): 554–78.
- Mascha, M. F., D. Janvrin, J. Plouff and B. Kruger. 2009. XBRL tools for small-to-medium-sized firms. *Strategic Finance* (January): 47–53.
- Phillips, M. E., T. E. Bahmanziari and R. G. Colvard. 2008. Six steps to XBRL. *Journal of Accountancy* 205(2): 34–9.
- U.S. Securities and Exchange Commission (SEC). 2009. *Interactive data to improve financial reporting – Rule 33-9002*. Washington, DC: Securities and Exchange Commission, January 30, 2009. Available online at <http://www.sec.gov/rules/final/2009/33-9002.pdf>.