



Extended Formalized Model for Financial Module of Open Source ERP System using REA Ontology

Jedpreeya Bekthaisong

Department of Software Engineering Faculty of Engineering, Chulalongkorn University
Bangkok, Thailand Email: jedpreya.b@student.chula.ac.th

Wiwat Vatanawood

Department of Software Engineering Faculty of Engineering, Chulalongkorn University
Bangkok, Thailand Email: wiwat@chula.ac.th

Abstract

The REA ontology is well-known as business ontology and it can be represented in various models such as ER Diagram, UML class diagram, etc. This paper demonstrates the reverse engineering of the Open Source ERP's source code into UML class diagram and performs the steps to classify the REA role of each particular classes of the ERP design model. We propose an extended meta-model of REA ontology which is written in UML class diagram. Our new meta-model helps to describe the original Economic Resource, Economic Event, Economic Agent and also the Support Event and its type. We found that our extended meta-model of REA works sufficiently well with our selected Open Source ERP product.

Index Terms: REA ontology, Open Source ERP

1. Introduction

Enterprise Resource Planning (ERP) is generally essential to the large organization. Since the ERP functionalities are difficult to understand (O'Leary, 2004) and the commercial or custom built implementation cost is expensive, there comes the alternative of Open Source ERP systems such as Opentabs, ADempiere or ERP5 which are also efficiently fit at lower cost or free of license fees. But how to choose the right Open Source ERP to work with, it is challenging. Using the design model of the software system is suggested to understand the system behaviors and easier to evaluate, it is also used as a generic model of the ERP system to be implemented. In this paper we reviewed the REA ontology, developed by McCarthy in 1982 as our reference structural design model. The REA ontology was firstly proposed to use data modeling techniques to systematically model business transactions and was used to assess the appropriate roles found in the accounting related software. It has been extended to Enterprise REA ontology (Geerts and McCarthy, 2000). There are a number of research papers proposed the relationships between ERP and REA ontology, but there is no such research that presents the relationship between Open Source ERP and REA ontology.

In this paper, we demonstrate such relationship between Open Source ERP and REA ontology. The selected Open Source ERP called ADempiere (ADempiere ERP, 2013) was used in the reverse engineering to obtain the class diagram, representing design model. Then, the resulting class diagram will be compared with the referred REA ontology. We demonstrate the comparison methodology and propose the extended version of the referred REA ontology in order to cope with the supporting activities found in the Open Source ERP system.

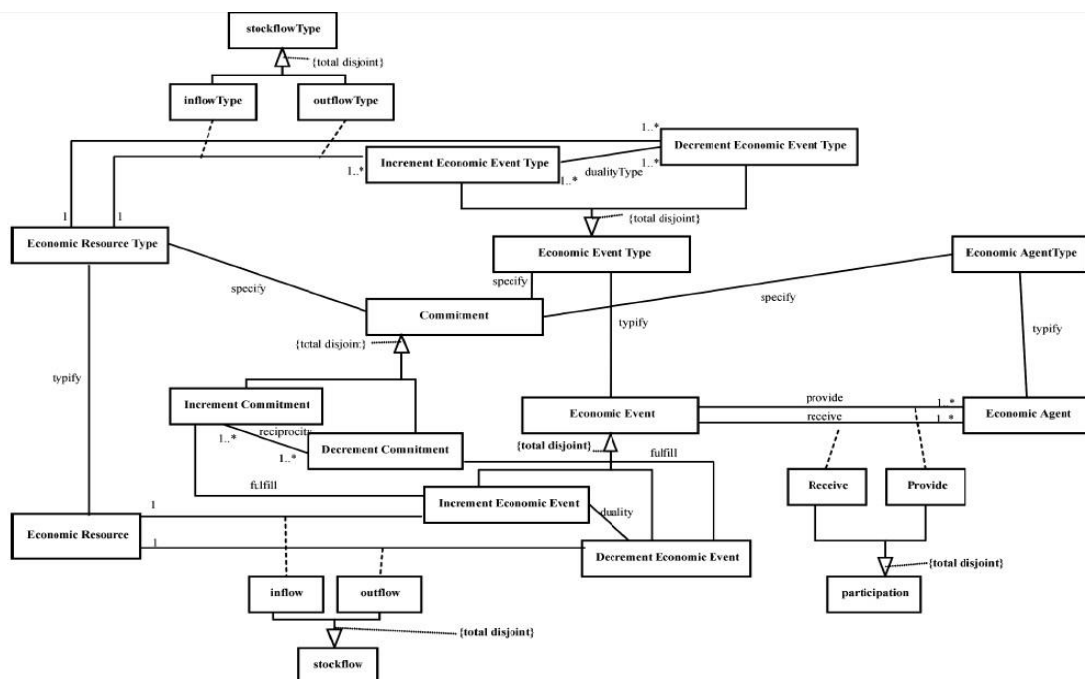
This paper is organized as follows. Section I is about the introduction. Section II describes the backgrounds on the REA ontology, ERP system and related work. Section III describes our comparison methodology. Section IV is our conclusion.

2. Backgrounds

Enterprise Resource Planning (ERP) is the huge integrated software system that gathers business processes and provides the functionalities to each department in the enterprise. The objective of ERP is to accumulate all of business processes as planning, production, sale, and marketing, human resource and accounting / finance together to provide the sharing database, report tools (Summer, 2004).

The ERP software products can be found in commercial ERP (SAP, PeopleSoft BAAN, Oracle, etc.) and Open Source ERP (Opentabs, Compiere, ERP5, etc.) (ERP Software 360, 2013). While, the REA ontology (Hruby, 2006) is the pattern of the conceptual design model to represent the enterprise business process for Accounting Information System (AIS). Each business transaction perspective was decomposed into Economic Resource, Economic Event and Economic Agent. There are research papers proposed the relationship between REA model and ERP software. For example, the relationships between REA Model and Data Model from commercial ERP product, called SAP was proposed by (O’Leary, 2004). The resulting consistent relationships were found among database, semantic, and structure orientation. The ontology-driven information system is now still on focus and the REA ontology is one of the alternatives among them. In object-oriented design and development, the original REA ontology was represented by UML class diagram (Sutheparaks et al., 2011) and successfully assured the completeness of the accounting and financial business transaction processing to the software design model. In addition, the redesign and formalization of the REA ontology was proposed in (Gailly and Poels, 2007) to be a referenceable meta-model, shown in Figure 1, of any REA model written in class diagram.

Figure-1. The meta-model of REA model in terms of class diagram (Gailly and Poels, 2007)



As shown in Figure 1, the Economic Resource, Event, and Agent are formally described and categorized. Both increment and decrement economic events are defined along with the appropriate commitments. Our approach to verify the successful Open Source ERP product, called ADempiere, with this REA meta-model in Figure 1. We intend to do the modification of this existing REA meta-model when it is needed after the verification.

3. Our Methodology

We simply perform three main steps as shown in Figure 2. The first step is to read the ADempiere source code and do the reverse engineering to get the corresponding raw UML class diagram. The second step is to classify each particular raw UML classes from step 1 into the class grouping table. An example of the classification of Asset part is shown in Table 1. By considering the role of each class described in REA meta-model along with the features specified in the ERP product manual, we are able to classify the classes in Resource column and their related classes in Event column. We also consider the classes in Agent column according to (Dunn et al., 2006)

Figure-2. Methodology steps

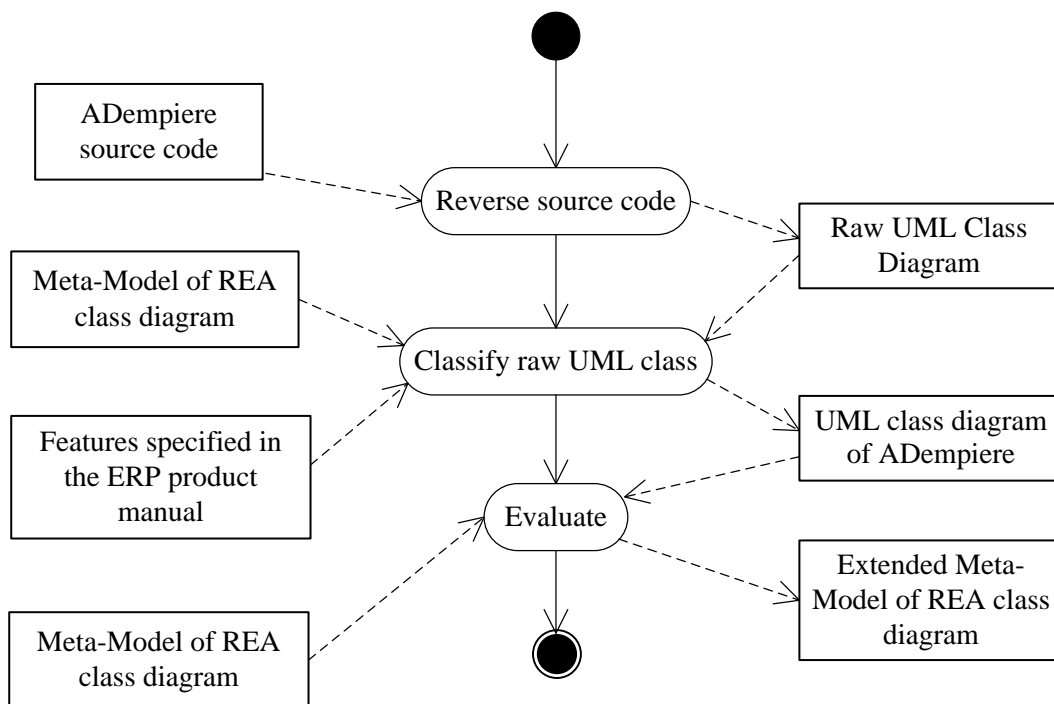
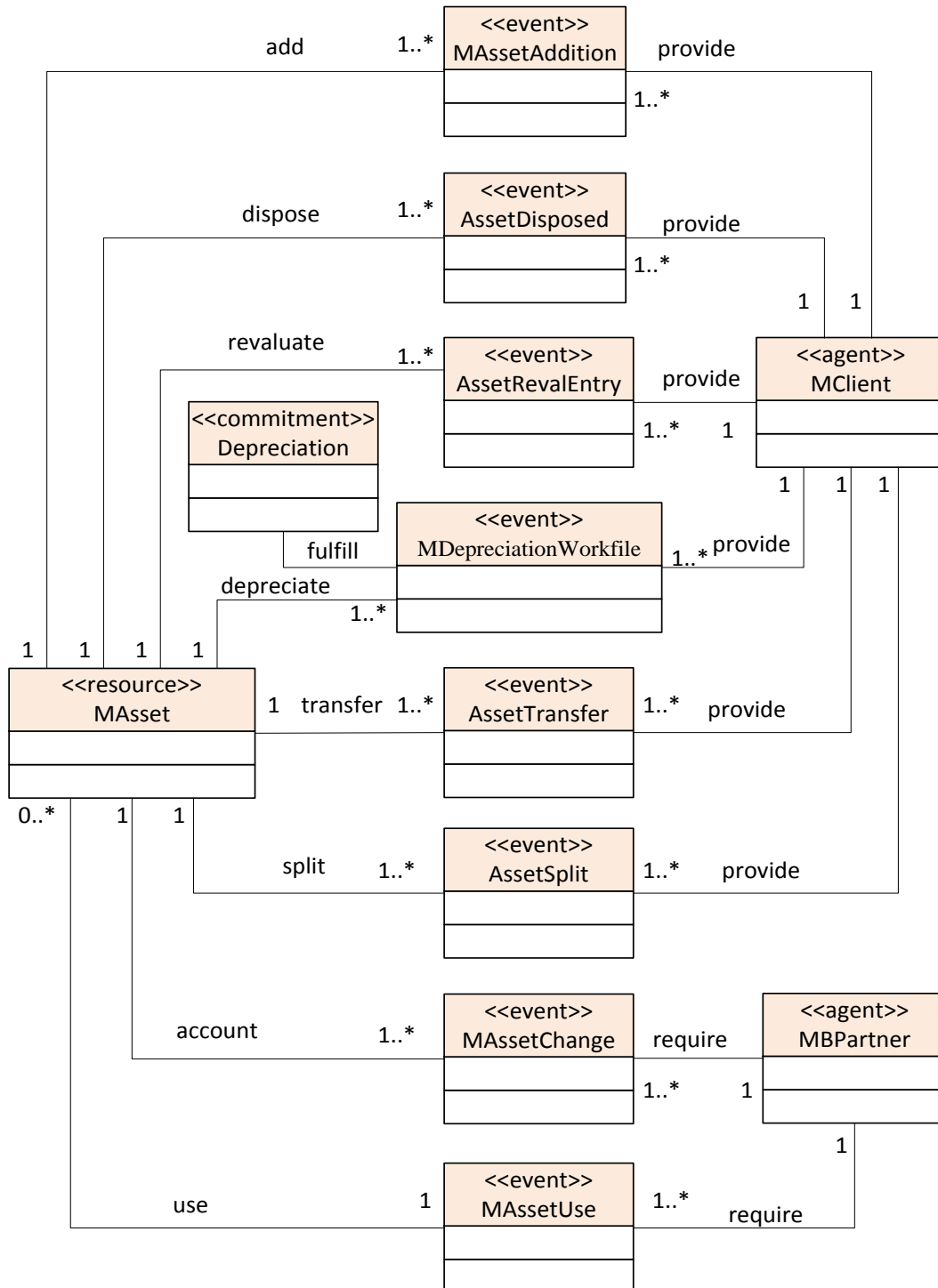


Table-1. An example of the Class Grouping Table for Asset Part

Resource	Event	Agent	Commitment
MAsset	MassetAddition	MBPartner	Depreciation
	AssetDisposed	MClient	
	AssetRevalEntry		
	MDepreciationWorkfile		
	AssetTransfer		
	AssetSplit		
	MAssetChange		
	MAssetDelivery		
	MAssetUse		

Figure 3 shows the UML class diagram of the asset part of ADempiere using REA ontology approach. The third step of our methodology is to evaluate the conformance between the resulting UML class diagram from step 2 and the referenceable meta-model in Figure 1.

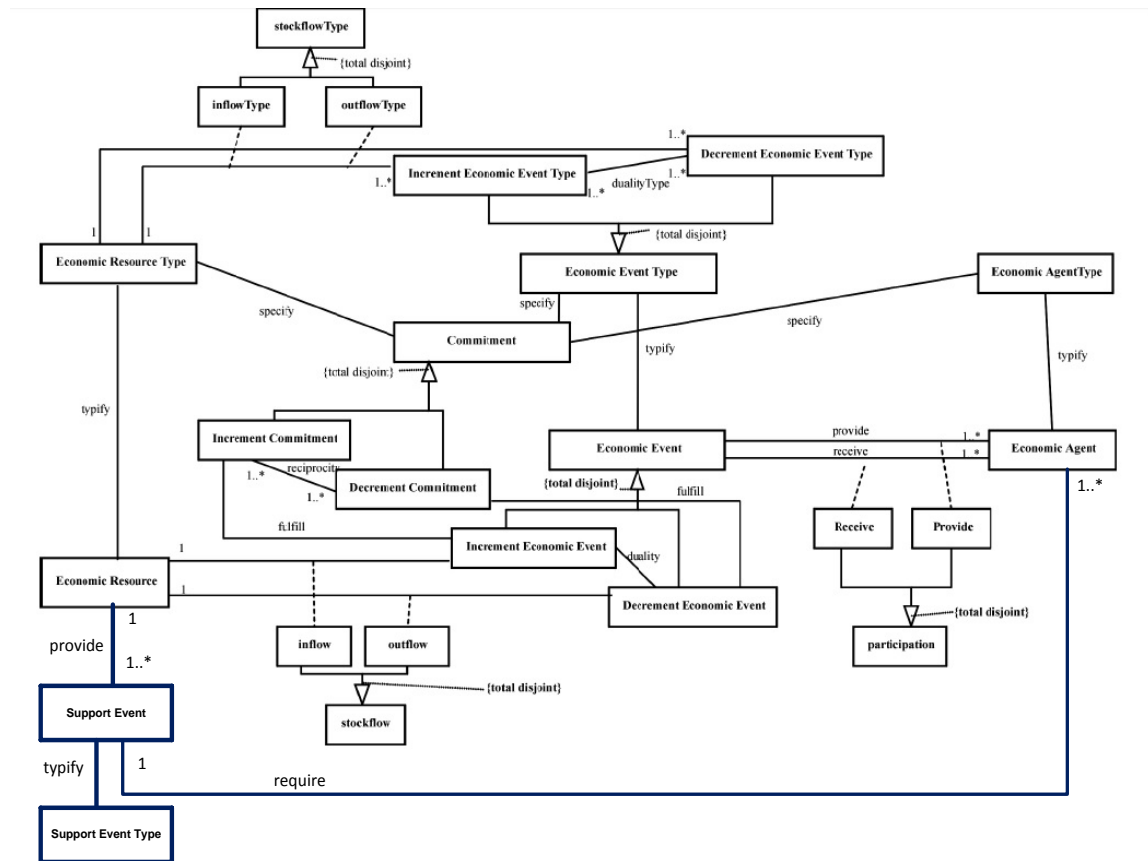
Figure-3. UML class diagram of ADempiere which is compiled as REA Ontology – Asset part



We found some related Event role classes - AssetTransfer, AssetSplit, MAssetChange, and MAssetUse, yield neither increased nor decreased values of the resource MAsset class. The specified event classes do just changing the resource attributes or logging the history of resource asset. These event classes are called Support Event, described in (Wongpinunwatana, 2012). The Support Event class helps control, plan and manage resource. For example, the AssetSplit class only splits resource asset into 2 asset entries, but the total value of both resource assets are the same value. Even more, the

AssetChange class only does the logging of the phenomena of resource asset without changing the asset value. We decide to extend the meta-model of REA class diagram in (Gailly and Poels, 2007) to cope with the Support Event and Support Type as shown in Figure 4.

Figure-4. Our Extended Meta-Model of REA class diagram



4. Conclusion

In this paper, the meta-model of REA class diagram in (Gailly and Poels, 2007) is extended to describe the Support Event and its Support Type. We found that our extended meta-model is more sufficiently describe the UML class diagram of the Open Source ERP product than the previous one. We select the Open Source ERP called ADempiere and have it reverse engineered into UML class diagram. The whole classes are walkthrough with our extended meta-model and we found that the Support Event and Type works well.

References

- ADempiere ERP. 2013 July 05. What is ADempiere [Internet]. Available from: http://www.adempiere.com/What_is_ADempiere%3F.
- Dunn, C. L., Cherrington, J. O., and Hollander, A.S. 2004. Enterprise Information Systems: A Pattern-Based Approach. Third ed. McGraw-Hill/Irwin. 554p.
- ERP Software 360. 2013 July 02. Top 5 Open Source ERP Software Applications [Internet]. Available from: <http://www.erpsoftware360.com/erp-open-source.htm>.
- Gailly, F., Poels, G. 2007. Towards Ontology-Driven Information Systems: Redesign and Formalization of the REA Ontology. The Accounting Review: 245-259.
- Geerts, G.L., McCarthy, W.E. 2000. The Ontological Foundation of REA Enterprise Information Systems. American Accounting Association Conference: 1-34.

- Hruby, P. 2006. *Model-Driven Design Using Business Patterns*. Springer-Verlag Berlin Heidelberg. 368p.
- McCarthy, W.E. 1982. The REA Accounting Model: A Generalized Framework for Accounting Systems in A Shared Data Environment. *The Accounting Review*: 554-578.
- O’Leary, D.E. 2004. On the relationship between REA and SAP. *International Journal of Accounting Information Systems* 5: 65– 81.
- Summer, M. 2004. *Enterprise Resource Planning*. Pearson Prentice Hall. 208p.
- Sutheparaks, U., Vatanawood, W., and Patanothai, C. 2011. Defining Global Schema for ETL of Human Resource Performance Appraisal System Using REA Ontology. Eighth International Joint Conference on Computer Science and Software Engineering (JCSSE): 275-280.
- Wongpinunwatana, N. 2012. *Financial and accounting systems for enterprise resource planning (ERP)*. College Arts Building, Don Bosco Technical College. 480p.