International Journal of Accounting Information Systems
5 (2004) 65–81

INTERNATIONAL JOURNAL OF
ACCOUNTING
INFORMATION
SYSTEMS

On the relationship between REA and SAP

Daniel E. O'Leary*

Marshall School of Business, University of Southern California, 3660 Trousdale Parkway, Los Angeles, CA 90089-1421, USA

Abstract

Resources–Events–Agents (REA) is the best known theoretical accounting enterprise database model, while SAP is the dominant enterprise resource-planning system. The purpose of this paper is to investigate some of the relationships between the underlying data models in REA and SAP. The criteria of Dunn and McCarthy [*J. Inf. Syst.* (1997) 31] for differentiating accounting systems (database orientation, semantic orientation, and structuring orientation) are used to structure a summary of those relationships.

Although it is found that there are substantial similarities, SAP does have some implementation compromises that generally keep it from being fully REA. These compromises are based on the use of accounting artifacts and other, often, implementation-specific compromises. In addition, there are emerging differences between the two.

A comparison between REA and SAP is important for a number of reasons. First, the tie between SAP and REA provides SAP with a theoretic basis. Second, the existence of a relationship between REA and SAP provides an important basis for the coverage of REA in the curriculum. Third, the opening of a dialogue between SAP and REA could spur further development of both.

Keywords: Theoretical accounting enterprise database model; REA; SAP

1. Introduction

Recently, enterprise resource planning (ERP) systems have gained substantial attention, providing firms with accounting models that are integrated with other activities of the firm, such as production planning and human resources. These ERP systems provide integration through the implementation of an enterprise database that spans the range of enterprise activities and locations.

* Tel.: +1-213-740-4856; fax: +1-213-747-2815.
E-mail address: oleary@rcf.usc.edu (D.E. O'Leary).

1467-0895/$ - see front matter
doi:10.1016/j.accinf.2004.02.004

ERP systems have been important to the growth of consulting. For example, reportedly, at one time, Price Waterhouse did roughly 40% of their consulting in the implementation of ERP systems (Public Accounting Reports, 1997). Among ERP systems, SAP has the largest market share (Busse, 1998). Furthermore, SAP has been adopted by large firms including a majority of the Fortune 500 (Busse, 1998). In addition, there is substantial information available about SAP in the popular literature (e.g., Busse, 1998, and others).

On the other hand, the dominant theoretical accounting enterprise database model is the resources–events–agents (REA) model developed by McCarthy (1982). That model has been analyzed and in some cases extended by Denna et al. (1994), Grabski and Marsh (1994), Hollander et al. (1996), Dunn and McCarthy (1997), Geerts and McCarthy (1997, 2002), Smith-David et al., (2002), and others.

1.1. Purpose

The purpose of this paper is to investigate relationships between the REA[1] model and the SAP data model. Increasingly, firms are turning to package software and not develop their own enterprise software. With an increasing emphasis on package software, models like REA generate value from their ability to help explain and understand such package software.

Dunn and McCarthy's (1997, pp. 36–37) criteria (database orientation, semantic orientation and structuring orientation) for differentiating accounting systems are used as the basis of summarizing the analysis. This paper finds that there are a number of similarities between the REA and the SAP data models. However, there also are apparent implementation compromises that limit the SAP model from being implemented as "pure" REA.

This paper does not attempt to provide evidence that SAP is based on REA, or conversely. Such an investigation would be virtually impossible. We can never know the original intent of SAP developers. In addition, systems evolve, and intent may change as other versions are presented. Furthermore, the development of a large software system is a team effort, with different team members having different design and implementation notions. As a result, SAP implementations are based on a portfolio of organizational decisions that will influence the implementation version relationship between SAP and REA.

1.2. Why examine this problem?

There are a number of reasons for focusing on the relationship between SAP and REA. Dunn and McCarthy (1997, p. 46) suggested that

> Potentially productive extensions in REA...research could include...use of REA to explicate...instantiations...The definition of this construct stems from the idea that a system consists of repeated occurrences of the structured REA template.

[1] The term REA is used because it generally is a more complete description of the contents of such databases. It is also meant to include REAL, although most of the research work has had to do with REA.

Previous research that has addressed issues regarding REA instantiations include Denna et al. (1994) and Grabski and Marsh (1994), who investigated the use of REA as a basis of manufacturing systems. However, there are no instantiations that focus on the relationship between REA and ERP. A comparison with SAP would provide one such basis.

Because SAP and other ERP models are commercial products, largely, market forces, rather than theoretic structures, have driven development and changes. As a result, a second reason for examining relationships between REA and SAP (and other ERP systems) is to provide a theoretical base to those ERP models. Such a theoretical base could provide a foundation that would facilitate anticipating future needs and changes, which may be ignored due to various implementation compromises (e.g., McCarthy and Rockwell, 1989).

In addition, the existence of similarities deriving from such a comparison suggests that there could be mutual benefits to both ERP, such as SAP, and REA. Similarities and differences could form the basis of a dialogue aimed at evolving each. Accordingly, a third reason for examining the relationship between REA and ERP models is to facilitate the additional development of both REA and ERP models, with differences feeding back to each other. Furthermore, ERP systems such as SAP are notoriously difficult to understand. REA provides one approach that can be used to facilitate the understanding of the underlying ERP model.

However, this investigation finds that REA provides a robust basis of system representation for ERP systems, such as SAP. This similarity provides a theoretical basis for SAP and a commercial "raison d'être" for REA, critical for education purposes. In addition, the comparison to SAP provides a number of potential extensions to REA structures, including more detailed organization models and a focus on tightening the links between data models and process versions. Finally, the comparison of REA and SAP data models brings into an SAP domain all of the long line of REA research for extensions and use in SAP.

This paper proceeds as follows. Section 2 provides a brief summary of ERP systems. Section 3 briefly summarizes the REA model. The heart of the paper is in Sections 4–8. Section 4 investigates the database orientation with compare SAP to REA. Section 5 analyzes the semantic orientation to compare SAP and REA. Section 6 summarizes the structuring orientation as a means of comparing SAP and REA data models. Section 7 briefly addresses the issue of analyzing the broader base of SAP models, by examining some issues deriving from the expanding REA's organization model to be consistent with SAP's capabilities. Section 8 analyzes the emerging use of "commitments" in both REA and SAP. Section 9 provides a brief review of the recent ontological analysis of REA. Section 10 investigates some of the compromises to REA often made as a part of SAP implementations. Section 11 summarizes the paper and discusses some extensions.

2. Enterprise resource planning systems

ERP systems provide firms with the ability to integrate across functional areas and operations. As an example, SAP, the ERP with the largest market share, has a number of different modules including financial management, controlling, treasury, project system, production planning, plant maintenance, sales and distribution, human resources, and materials management. Rather than functioning as isolated or independent functional or operational activities, these modules are integrated so that when one piece of information is updated, each application has updated information. That single database with multiple views is generally regarded as one of the most important aspects of the system.

Some of the better known ERP systems besides SAP include, PeopleSoft, Baan, and Oracle. These systems typically are integrated with a relational database, such as Oracle, IBM's DB2, Informix, or Microsoft's SQL.

Firms implementing ERP systems can employ either processes generated internally or from the Big Four, or they can draw on best-practice processes available in ERP software. In general, ERP systems employ classic accounting processes as a part of their best-practice portfolios. As a result, there is a focus on traditional accounting artifacts, such as the general ledger, despite that theoretical accounting enterprise models, such as REA, do not require such artifacts.

2.1. SAP

SAP offers those best practices structured in over 10,000 different models. In SAP, those best practices are implemented via tables and options in those tables.

SAP's reference model has six submodels (Curran and Keller, 1998, pp. 42–43 and 250–258): interaction model, business object, data model, process model, organization model, and business application component model. The interaction model illustrates the primary organizational units exchanging information for different business activities. The business object model allows the modeling of business rules. Business objects contain the data models, so that users can navigate through the data models to the necessary level of detail. The data model illustrates what information capabilities SAP R/3 supports. Additional information regarding the data model is given as part of the analysis in Sections 4–6. The business process model captures process interaction and consequences of events. The process model also contains a value chain view, indicating how process chains connect to each other. The process focus means that there is less emphasis on classic functional silos than with other accounting models. An organization model is used to represent the organization structure and to capture organizational interactions. A customizable workflow model is associated with the organization model. The workflow model integrates the personnel and workflow with tasks. The business application component model indicates how different components interact.

3. REA data model

The REA model (McCarthy, 1982) is the generally accepted theoretical accounting enterprise model in accounting information systems. Much of the extensive literature growing up around REA is summarized in Dunn and McCarthy (1997) and Smith-David et al. (2002).

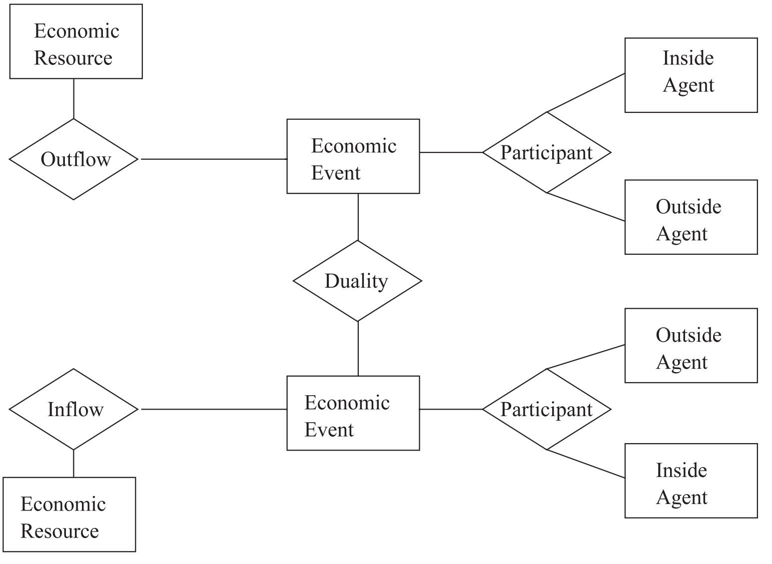

Fig. 1. REA model. Source: McCarthy (1982).

The use of REA has taken a number of directions. Over the years, the REA model has been used in a number of ways. Hollander et al. (1996) treat it as a data model and use an extension of it as an enterprise modeling approach. Furthermore, McCarthy's (1982) formulation has been and continues to rely heavily on the theoretical work of Ijiri (1975). In particular, the original model was largely an ER model representation of Ijiri (1975). Subsequent work, such as commitments (see below), also draws heavily from Ijiri (1975). In addition, Geerts and McCarthy (2000, 2002) extend the basic model and develop ontological foundations of REA, using it as a theory or metatheory.

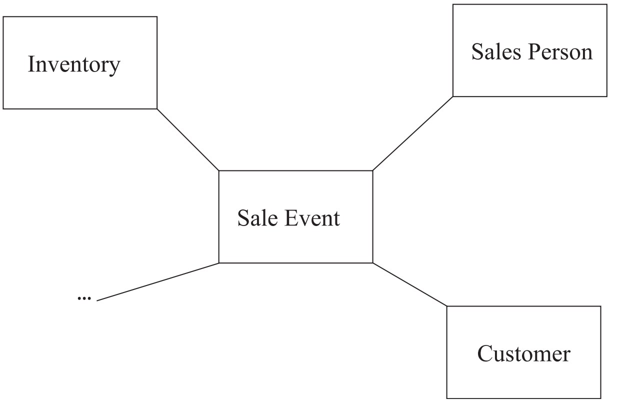

Fig. 2. Sales event using REAL diagram.

Hollander et al. (1996) extended the model to include location, using the revised name REAL. In addition, there have been a number of example REA system prototype implementations, such as Denna et al. (1994) and Grabski and Marsh (1994).

In the literature, REA generally is couched as a data model, with much of its initial REA representation (McCarthy, 1982) structured in entity–relationship diagrams and its early implementations in relational database format (Chen, 1976). An entity–relationship model version of REA is summarized in Fig. 1. This template is usually employed to model at the process level.

In addition to putting location fully in the name REA, Hollander et al. (1996) also present a slightly different form of diagrammatic representation that they referred to as REAL models, which basically include only entity forms. A sample REA(L) model is summarized in Fig. 2. Generally associated with events are resources (inventory), agents (sales person and customer), and locations (none in this particular example).

4. Database orientation

As noted by Dunn and McCarthy (1997, p. 36).

A database orientation as defined here has three conditions:

1. data must be stored in their most primitive levels (at least for some period),
2. data must be stored such that all authorized decision makers have access to it, and
3. data must be stored such that it may be retrieved in various formats as need for different purposes.

SAP's data orientation is consistent with this database orientation. The data are captured in a relational model (most frequently Oracle) and stored for a period of time. Decision makers are given access to data through reports and forms. In addition, for those situations where the SAP reports do not meet the needs of the decision maker, the underlying data model can be queried directly. In some settings, to facilitate that ability to query, the data are passed on to a data warehouse (e.g., SAP's business information warehouse), facilitating the analysis of data over periods greater than a single year (O'Leary, 1999).

5. Semantic orientation

Dunn and McCarthy (1997, p. 36) note that semantic orientation, "...means that all potential users of a database pool their notions of information concepts and use that integrated set of ideas to build one conceptual data model that serves everybody". Dunn and McCarthy (1997, p. 37) further note that "...components of the models should reflect real world phenomena, a situation that precludes use of basic double-entry artifacts...as declarative primitives". The entities and the events being modeled are critical aspects of the REA model, as is its ability to capture cross-functional enterprise information flow. Although the SAP model has many similarities with REA, there are some implementation compromises.

5.1. SAP data model: REA entity compliant

Because one of the primary concerns of the REA models is the data model, that is our focus of comparison between SAP and REA.

> The (SAP) data model analyzes how information objects, that is data, interact with preceding and following functions within the business process model. The data model illustrates the information input needed to perform a given set of tasks. This data viewpoint depicts the most important objects and describes them and their relationships with one another. To successfully perform a task, data input must first be received from preceding tasks (Curran and Keller, 1998, p. 40).

Thus, within the SAP data model, events are linked to each other, providing further information. In addition, within the SAP model, information is identified as needed to support those events. As we will see in the following example, multiple events are linked to multiple different resource, agents, and locations.

To facilitate comparison, a schema will be called *REA entity compliant* if the database has entities of the type that are resources, agents, events, and locations. We use the term compliant because it used to establish whether SAP is compliant with REA.

Carpark example. An example will illustrate a SAP data model and compare it with an REA structure. Consider the following example data model from Curran and Keller (1998, p. 38).

> The Carpark data model includes the key information objects 'workshop/garage,' 'customer,' 'product/material,' and 'vendor.' The 'workshop/garage' is a central object that accounts for much of the added value of the business process. The customer and vendor are the most important units with which the dealership has a business relationship. 'Product/material' represents Carpark's main business area, that is, the cars, spare parts, etc. In which the company deals. Fig. . . . (3) . . . illustrates Carpark's business information objects.
>
> There are several types of customer: the orderer, the goods receiver and the debtor.

Fig. 3 provides a representation of the carpark example, represented as a SAP data model based on Curran and Keller (1998). To facilitate the comparison to see that the SAP representation is an REA entity compliant, the descriptors, resources, events, agents, and locations have been added to the figure. This example illustrates for this well-known example; SAP appears to be consistent with REA(L) and is REAL entity compliant.

Fig. 3 could be reconstructed as REA diagrams with different levels of detail in any of events, agents, and resources. As an example, the REA diagram might have three events (purchasing, delivery, and car repair) or it could break down specific events in more detail. The purchase event could consist of three interrelated events: customer inquiry, order, and offer. Similarly, agents could be represented in different levels of detail.

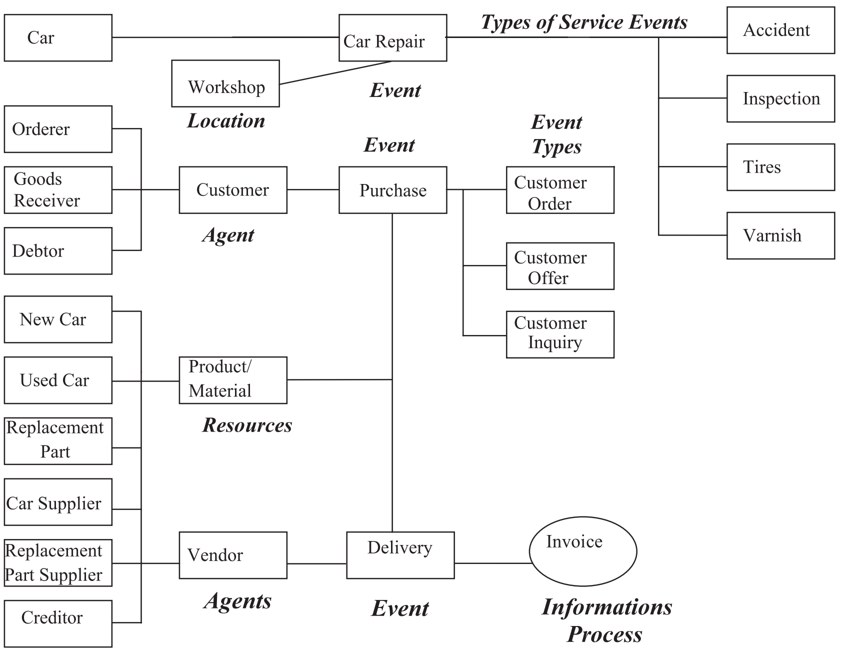

Fig. 3. Partial SAP example data model. Source: Curran and Keller (1998, p. 39).

5.2. Events and triggers

SAP's semantic orientation could be characterized as event focused and corresponding input document focused because the documents capture the information surrounding particular events. There is a relationship between economic events that would be captured in an REA setting and the types of activities that ultimately trigger changes to the SAP-based database. For example, as noted in Whang et al. (1995), the following activities trigger changes in the materials management module:

- Receipt of goods
- Use of materials
- Transfer of goods and materials (plant to plant moves)
- Finished goods movements (parts out of a warehouse and goods into a warehouse).

Each of these could also be an event in a REA representation.

5.3. Cross-functional design

Furthermore, neither SAP nor REA are designed using functional silos. Instead, both SAP and REA model the enterprise, focusing on its many events. Both SAP and REA are enterprise models integrating financial, production, and other aspects into the same database.

5.4. Implementation compromises

However, SAP's semantic orientation is not pure. SAP also employs a double-entry accounting structure, including such artifacts as a chart of accounts. As noted by ASAP (World Consultancy; 1996), SAP includes a core of subsystems that interface with the general ledger system. Furthermore, each SAP implementation employs multiple best-practice processes. The choice of those processes may also result in non-REA implementation compromises.

6. Structuring orientation

As noted by Dunn and McCarthy (1997, p. 37), "a structuring orientation mandates the repeated use of an occurrence template as a foundation or accountability infrastructure for the integrated business information system". REA structure manifests itself in at least two dimensions: duality and templates.

6.1. Duality

Perhaps, the most intriguing characteristic of REA is duality. According to McCarthy (1982, p. 562).

> Duality relationships link each increment in the resource set of the enterprise with a corresponding decrement... Increments and decrements must be members of two different event entity sets: one characterized by transferring in (purchase and cash receipts) and the other characterized by transferring out (sales and cash disbursements).

There is evidence of duality within SAP. For example (Whang et al., 1995), within SAP, customer inquiries lead to quotations, which lead to sales orders, which lead to delivery. In this setting, inquiries are dual to quotations, which are dual to sales orders, which are dual to deliveries.

6.2. Object-oriented models and REA compliance: template structure

REA was originally cast in a relational database format, exploiting the entity–relationship model of Chen (1976). However, since REA's development, object-oriented forms of database representation have become increasingly important. In addition, SAP can exploit object-oriented database structures.

It is unlikely that SAP was created through repeated template application. However, there is evidence that SAP models can be represented in REA templates.

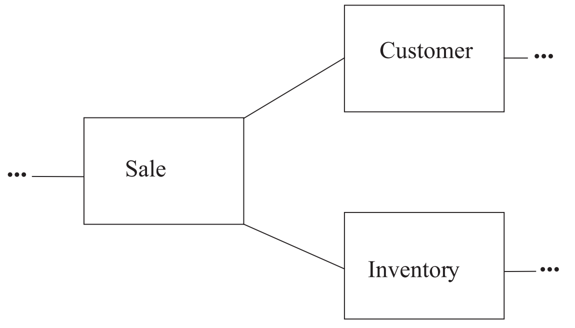

Fig. 4. SAP sale event (based on Whang et al., 1995).

Chen et al. (1995) introduced the notion of REA compliance to guide a schema evolution of an object-oriented accounting system. As noted by Chen et al. (1995, p. 494).

> ...REA classes are meta classes. A class of an accounting database is an instance of one of the classes of the REA model...A family of classes in an accounting database will be said to be REA-compliant if and only if
>
> - the class is an instance of one of the REA meta classes
> - it inherits all the attributes from this REA meta - class
> - the values of these attributes of the class are defined
>
> For example, the class Purchase Payment is an instance of the meta class Duality. The class inherits the attributes, increment and decrement, from the meta class duality. Furthermore, its increment part is defined as purchase and its decrement part is defined as Cash Disbursement. Both of them are events. Hence, Class Purchase Payment is REA-compliant. If all classes of an accounting database are REA-compliant, then the schema of this accounting database is REA-compliant.

This definition is largely structural. Intuitively, Chen et al. (1995) have defined a schema as REA compliance if it can be represented as a set of related REA templates. It is this definition that will be adopted here, extended to include location and labeled REA compliance.

To illustrate REA compliance, we present an example that derives from information about SAP flows and documents as presented in Whang et al. (1995). Whang et al. (1995) present a number of SAP screens including sample data provided by SAP about a sales event . Using those screens, we can create the underlying necessary relational data model that is required to support those screens. A partial representation of the sales event is summarized in Fig. 4. In this case, we can see that SAP is REA compliant because the template for the sale in SAP contains the same object information (Fig. 2).

7. Other SAP models: SAP's organization model

The broad base of integrated SAP models makes additional demands on the underlying data model that have not yet been made on the REA model. Although academics may some day generate more complete REA models, there are some gaps in the REA model. Because research progresses by identifying gaps and filling them, this section identifies one such set of unfulfilled REA models by focusing on the SAP organization model.

Comparison of REA and SAP does yield an understanding that the basic organization model in SAP, generally appears to have additional capabilities than that described by REA. Although REA (McCarthy, 1982) can generate organization relationships through a recursive data model, generally, tasks are not included in the REA model. Geerts and McCarthy (2000, 2002) show generic links between tasks and REA; however, they are not specified but are presented as a generic link. The inclusion of tasks would not be hard to model, but could broaden the REA model. Tasks could then be captured as event information. Such task information could provide important control information for on-line audit activity. In addition, task information could provide quality control check information on individual events, e.g., for completeness. Furthermore, event task information could provide a basis for process improvement and reengineering.

As an example, consider the organizational enterprise data model example from ASAP (World Consultancy; 1996, p. 175), as illustrated in Fig. 5. The model employs entities defined as a part of REA, including agent (person) and locations (organizational unit and workplace).

However, the model provides additional capabilities beyond REA. Generally, task and job are not directly cast as part of REA models; however, that may be a function of the particular instantiations previously generated as REA models. REA models of production processes would likely need such information. Workflow information, generally, has been ignored in REA instantiations developed to-date.

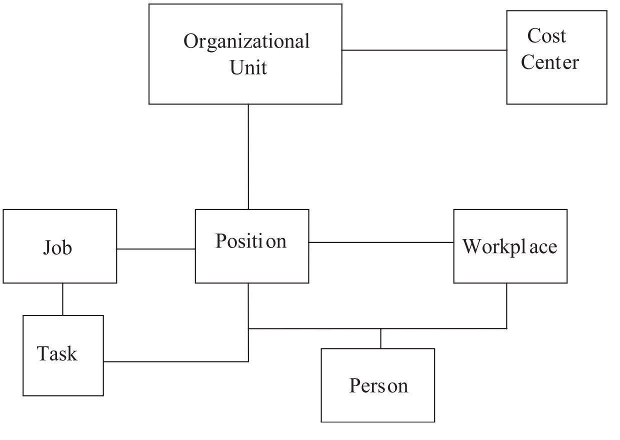

Fig. 5. Part of enterprise data model. Source: ASAP (World Consultancy; 1996, p. 175).

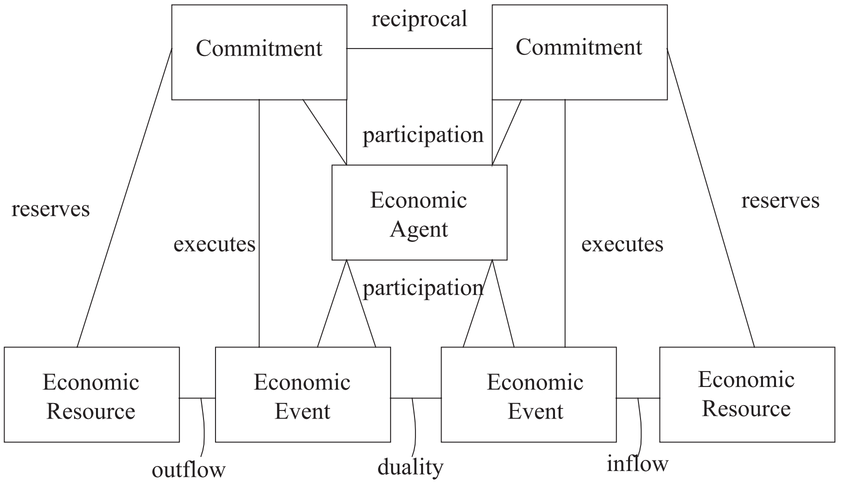

Fig. 6. REA with commitments.

8. Extensions of REA to commitments

Recently, Geerts and McCarthy (2000, 2002) have presented a revised version of REA that adds commitments to the REA model, based on the model of commitments of Ijiri (1975), as seen in Fig. 6. Commitments are defined by Ijiri (1975, p. 130) as an "agreement to execute an economic event in a well-defined future that will result in either an increase of resources or a decrease of resources". Geerts and McCarthy (2000, 2002) differentiate between two types of events, a transfer and a transformation: "A transfer executes a contract while a transformation executes a schedule. For example, a sale executes a sales order, and a production job executes a production order". In addition, they include the notion of reserves, which is a special type of stock flow relationship. As noted by Geerts and McCarthy (2000), "a sales order results in a reservation of the finished goods to be delivered, while a production order results in a scheduled completion of goods".

8.1. Ijiri (1975) on commitments

With the development of commitment accounting, Ijiri (1975) was particularly concerned with issues such as the "presentation of the firms financial status on the balance sheet". As a result, he listed additional accounts that would appear under commitment accounting, including the following artifacts:

- Commitments receivable
- Inventories on order
- Services contracted for

- Fixed assets contracted for
- Commitments payable
- Sales orders outstanding
- Disposals outstanding

Ijiri (1975) thought that performance measurement could be improved using commitments. In particular, he noted that profit recognition, at the point of commitment, could be discussed separately.

Commitment accounting, as discussed by Ijiri (1975), is not a part of the generally accepted accounting principles (GAAP). As noted by Ijiri (1975, p. 134), "...commitments...are not recognized in accounting unless they are half executed...".

8.2. SAP on commitments

SAP, the company with the ERP system with the largest market share, has long allowed the concept of commitment (http://help.sap-ag.de/sapdocu/core/470/helpdata/EN/35/2cd77bd7705394e10000009b387c12/frameset.htm), and so, is consistent with non-GAAP properties of REA.

Commitments can be used in both the financial accounting and controlling modules. For financials, the commitments and liabilities include:

- Outstanding orders (delivery commitment from confirmed orders)
- Open purchase orders (acceptance commitment from binding orders)
- Bill of exchange liability (total bill of exchange commitments at a bank)

Commitments are contractual or scheduled expenditures not yet reflected in financial accounting but that will lead to actual expenditures in the future.

SAP notes that commitment management provides an early recording and analysis of such commitments for their cost and financial effects. In the CO module, commitments can be entered for the following:

- CO production orders
- Production orders
- Internal orders
- Maintenance orders
- Sales orders
- Cost centers
- Projects (work breakdown structure elements)

9. Ontological foundations of REA and SAP

ERP systems are, by necessity, well specified because they are computer programs. As a result, ERP systems like SAP have thousands of terms that are defined (e.g., http://

help.sap-ag.de/sapdocu/core/470/helpdata/EN/35/2cd77bd7705394e10000009b387c12/frameset.htm). For example, there are about 750 terms in the "t's," and over 1500 terms in the "p's." As Geerts and McCarthy (2000, 2002) flesh out the REA ontology, there are some emerging ontological differences between REA and SAP. For example, Geerts and McCarthy (2002) note.

> Essentially, tasks are REA compromises where some occurrences in time are important enough to be specified but not important enough to be represented individually and tracked. Tasks in component form are excluded from the ontological analysis that follows..., because a *full-REA* representation would not need them. This is perhaps a point that needs to be debated in future ontological analysis of this model. Such a revision could go either way.

However, within SAP, tasks are a critical aspect of the model. Tasks are something to be managed, controlled, and reported on. Bundles of tasks are assigned together to resources at any one time to facilitate planning. Furthermore, tasks have owners, and tasks are part of groups of tasks.

10. REA compromises in SAP

As noted by McCarthy and Rockwell (1989), inevitably, there will be compromises in REA implementations, "...an events accounting system is a theoretical ideal which Realistically would never be implemented". They note that unless storage became costless and abstracting detail was "painless", there would never be full event histories. As a result, they implemented a number of implementation heuristics in a tool that they designed to provide heuristic guidance. Those implementation compromises included the following:

- temporal aggregation of event histories, where detailed transaction histories are not needed;
- representation and use of a subset or superset of the entity information; and
- nonimplementation of parts of the REA model, e.g., internal agents.

SAP is likely to contain a number of implementation compromises. Embedded in SAP are classic artifacts, such as the general ledger, that are required for traditional accounting system operation. One rationale for the existence of these non-REA structures is that their use is market driven. Historically, accounting systems have been artifact based, and the people using output from accounting systems have used accounting artifacts. Second, each of the implementation compromises discussed in McCarthy and Rockwell (1989) may be implemented in any SAP implementation because a company implementing SAP must chose between different best practices for its specific implementation. Theoretically, those best practices would be chosen to be cost beneficial. Thus, potentially, we would anticipate all three implementation compromises elicited by McCarthy and Rockwell (1989) and, probably, additional actual or virtual compromises as well, particularly relating to specific accounting artifacts.

11. Differences between REA and SAP

When compared with ERP systems like SAP, REA has many similarities, as we have seen in this paper. However, there are also differences.

SAP is a highly specified computer software, but REA is more of a template that can be used to understand software. As result, as noted earlier, SAP has defined thousands of terms as part of its enterprise software, whereas REA has defined an enterprise template that can be applied to particular situations or can be matched to existing settings, as done in this paper.

That greater specificity in ERP systems has lead to a number of implementation compromises such as the use of classic debit–credit accounting systems with a general ledger. The compromises could well be market driven, but are compromises, nevertheless. One might argue that data classification into a general ledger would correspond more closely to adherence to functional silos, rather than processes, but that criticism is arguable because the classification occurs in the context of the process focus of SAP.

On the other hand, the ERP system specification provides definitive process and task definitions. For many real-world settings, REA is underspecified. For example, if we want to know how a human-resources process works, REA provides no direct insights. However, given a human resources model or system, we can map it to REA to try to understand it better or we can build a system using REA as a guide to the underlying data model, etc.

REA generally is broader than ERP systems are. However, the lack of specificity inhibits actual use in a real-world setting, without substantial system design and implementation, typical of the world that proceeded package software. However, REA still provides us with an ability to understand existing software and processes and provides a model for future system design beyond classic general ledger systems.

However, in some ways, ERP systems are broader than REA. Although both REA and SAP have their origins as accounting systems, SAP has long been associated with other aspects of the firm, including production–planning functions that have just begun to be addressed in REA research (e.g., Verdaasdonk, 2003). Furthermore, systems like SAP have addressed issues such as human resources that have received little, if any, attention in accounting systems.

12. Summary, contributions, and extensions

This section briefly summarizes the paper, discusses some of the contributions of the paper, and investigates some extensions.

12.1. Summary

SAP was found to be consistent with REA in its database, semantic, and structure orientations. However, there were some implementation compromises in the structuring and semantic orientation of the SAP data model. There apparently are emerging differences between SAP and REA, e.g., at the task level.

12.2. Contributions

This paper has a number of contributions. First, REA is apparently quite robust in its ability to represent SAP data models. As a result, that relationship can be used to establish a theoretical basis of SAP and other ERP models. Second, the clear commercial importance of ERP models provides a readily apparent link in the importance of studying REA models. Third, this paper provides an additional instantiation of REA models that Dunn and McCarthy (1997) indicated was so important to pushing the frontiers of REA research. Fourth, the opening of a dialogue between SAP and REA might help spur further development of both.

12.3. Extensions

SAP has a number of different representations including (Curran and Keller, 1998, p. 42) interaction model, business object, data model, process model, organization model, and business application component model. Future research could focus on these other models and how they relate to REA representations. For example, in general, there has been little discussion of direct integration of REA with processes that they represent, except for those inherent in its database structure. As another example, there are no discussions that the authors are aware of the directly related REA entity relationship models and data flow diagram models or flow chart models. However, there has been a loose linkage between processes and data models discussed for some REA processes (e.g., Hollander et al., 1996).

Furthermore, there has been limited attention given to integration of REA with value-creation evaluation efforts, such as those models promulgated by Porter (1985). One view in the process model is that of a value chain, which consists of the most important process chains, how they are connected and how they are used to represent value in the company (e.g., Keller, 1996). These approaches can be integrated with REA efforts, such as in Geerts and McCarthy (1997).

In addition, SAP has the ability to use a wide range of process best practices. As a result, future research could focus on additional processes available within SAP. Additional analysis might focus on specific best practices or the general intent of SAP.

Furthermore, ERP and supply chain systems can be the basis of further analysis using REA. This work has been initiated by Haugen and McCarthy (2000). Other areas, such as customer relationship management, can also be investigated.

Finally, although this paper has focused on the relationship between SAP and REA, the analysis of relationships could be extended to other ERP systems including PeopleSoft and Oracle Applications, etc.

Acknowledgements

An earlier version of this paper was presented at SMAP in New Orleans in August 1998. I would like to acknowledge the comments of Julie Smith David and William

McCarthy on that version of this paper. I would also like to thank the reviewers of the paper for the American Accounting Association meeting, San Diego, August 1999, and for comments at that meeting. Portions of this paper were presented in Brisbane Australia, December 2000, as part of the International Research Symposium on Accounting Information Systems. Finally, I would like to thank Stewart Leech for his comments on the most recent version.

References

ASAP (World Consultancy). Using SAP R/3. Indianapolis (IN): Que; 1996.

Busse T. SAP to broader access to R/3, Computerworld, September 15, 1998.

Chen P. The entity relationship model—toward a unified view of data. ACM Trans. Database Syst. 1976;9–36 [March].

Chen N, McLeod D, O'Leary D. Schema evolution for object oriented accounting database systems. Expert Syst Appl 1995;9(4):591–8.

Curran T, Keller G. SAP R/3 business blueprint. Upper Saddle River (NJ): Prentice-Hall; 1998.

Denna E, Jasperson J, Fong K, Middleman D. Modeling conversion process events. J Inf Syst 1994;43–54 [Spring].

Dunn C, McCarthy W. The REA accounting model: intellectual heritage and prospects for progress. J Inf Syst 1997;31–51 [Spring].

Geerts G, McCarthy W. Modeling business enterprises as value added processes with resource–event–agent object templates. Paper presented at SMAP, Dallas, TX; 1997.

Geerts G, McCarthy W. The ontological foundations of REA enterprise information systems. Unpublished paper; August 2000.

Geerts G, McCarthy W. An ontological analysis of the economic primitives of the extended REA enterprise information architecture. Int J Account Inf Syst 2002;3(1):1–16.

Grabski S, Marsh R. Integrating accounting and manufacturing information systems: an ABC and REA-based approach. J Inf Syst 1994;61–80 [Fall].

Hollander A, Denna E, Cherrington O. Accounting, information technology and business solutions. Chicago: Irwin; 1996.

Ijiri Y. Theory of accounting measurement. Sarasota Florida: American Accounting Association; 1975.

Keller G. Beating a path through the technology jungle. SAP Info D&T, No. 49/50; June 1996.

McCarthy W. The REA accounting model: a generalized framework for accounting systems shared data environment. Account Rev 1982;554–77 [July].

McCarthy W, Rockwell S. The integrated use of first order theories, reconstructive expertise, and implementation heuristics in an accounting information systems. Proceedings of the ninth international workshop on expert systems and their applications; 1989. p. 537–48. Avignon, France.

Haugen R, McCarthy W. REA, a semantic model for Internet supply chain collaboration. http://jeffsutherland.org/oopsla2000/mccarthy/mccarthy.htm.

O'Leary D. REAL-D: a schema for data warehouses. J Inf Syst AAA 1999;13(1):49–62 [Spring].

Porter M. Competitive advantage. New York: The Free Press; 1985.

Public Accounting Reports. How do the system integrators rate? Public Accounting Reports; March 15, 1997.

Smith-David JS, Gerard GJ, McCarthy WE. Design science: an REA perspective on the future of AIS. In: Arnold V, Sutton SG, editors. Researching accounting as an information systems discipline. Sarasota, Florida: American Accounting Association, 2002. 35–63.

Verdaasdonk P. An object-oriented model for ex ante accounting information. J Inf Syst 2003;17(1):43–61 [Spring].

Whang S, Gilland W, Lee H. Information flows in manufacturing under SAP R/3. Stanford University Graduate School of Business Case OIT-13; June 1995.