The REA Accounting Model as an Accounting and Economic Ontology
(Version 0.90 -- 2019)

By

William E. McCarthy, Michigan State University
Guido L. Geerts, University of Delaware
Graham Gal, University of Massachusetts

The difficulty lies, not in the new ideas, but in escaping from the old ones, which ramify, for those brought up as most of us have been, into every corner of our minds

John Maynard Keynes (1936, p. viii)

Please note that this document is an authors' draft (version 0.90). We have many small additions and corrections here that are yet to be made, but this 0.90 version conveys, in a very substantial way, the essential monograph contents. We make this version available to accelerate dissemination of our ground-breaking content.

The finished version of the monograph is available from the American Accounting Association (AAA) bookstore:
https://aaahqbookstore.org/catalog/book/rea-accounting-model-accounting-and-economic-ontology
The official AAA version has all the corrections and content, plus the final figures with detailed color on many.

Preface

When the original Resource-Event-Agent (REA) paper was published (McCarthy W. E., 1982), its subtitle – a generalized framework for accounting systems in a shared data environment – signaled an attempt to unite the modeling of enterprise economic phenomena from two very different perspectives. These were: (1) the viewpoint of traditional accounting data users, and (2) the viewpoint of non-accounting decision-makers in an enterprise. At that time, those differing perspectives were largely served by different classes of computerized business systems, and it was McCarthy's intent to unify those views in one, explicitly-semantic scheme. ***Interoperability***[1] of information systems was both REA's key conceptual vision and its driver for continued development.

In 1982, database systems with object-oriented capabilities were not yet ascendant, and the company-wide computing platforms of the ERP (enterprise resource planning) era were just starting to be formulated, so within-company interoperability capabilities were limited technologically. And because large-scale implementations of the Internet and the World Wide Web were still 8-10 years distant, electronic commerce transactions between different firms were restricted to private, syntactically-coded EDI (electronic data interchange) networks, within which market exchanges were differently and redundantly captured by the computer systems of both trading partners. Blockchains of course were a far distant vision for any kind of an information system. Overall in 1982, interoperability was an idea that was wonderful in theory, but somewhat restricted in practice.

Today, most of those technological barriers to the REA interoperability vision have been lifted, but multiple process obstacles remain, many of them directly related to traditional accounting artifacts. As indicated at the outset of the 1982 paper, these double-entry system obstacles include:

1. its limited dimensionality,
2. its oft-times inappropriate classification schemes,
3. its too-coarse aggregation levels, and
4. its general non-integrative approach.

As noted independently in 2012 by a task force developing a conceptual approach to managerial costing (The Institute of Management Accountants, 2012, p. 59), all of these process barriers remain in force today, having survived both the interoperability revolution and the business process reengineering movement. Legacy complications still persistently bedevil accounting systems.

The double-entry system dates back over 700 years, and it was codified in published form in 1494 by Luca Pacioli (Geijsbeek, 1914). Over the ensuing centuries, and especially as accelerated by the industrial revolution, it has accumulated a large stock of engineered artifacts that helped accommodate (in a paper document age) the growing complexities of large-scale commerce. Some of these engineered artifacts – like periodic and absorption

---

[1] Formally, interoperability is the ability of data that is created in one information system to be used appropriately in another information system. The scope of interoperability can range from between-systems to between-enterprises to between-trading communities to the Internet (Obrst, 2003).

costing schemes – were certainly present at the dawn of the business computer age during the 1960s, and their unexamined wholesale adoption into the newer business information systems was akin to the clichéd mismatching blunder of "paving the cowpaths." Additionally, the development of new computer tools – most especially, pervasively useful tools like the spreadsheet – actually encouraged the introduction by accounting-minded practitioners of additional dysfunctional double-entry artifacts, among them the accelerated use of facet-coded accounts in computerized general ledgers. These new coding-conventions led to exploding charts of accounts, dysfunctional code-blocks, and wildly-mismanaged, non-interoperable semantics.

It is into this environment that we attempt to place the REA model today. Its progress in almost 40 years of affecting accounting practice, accounting teaching, and accounting research has been slow but steady.

a. **Practice**. In actual practice, there are software systems developed by firms like Workday, IBM, and REA Technology that explicitly place REA representation principles at the foundation of their enterprise software architectures. Workday (Nittler, 2018) has been especially forward in its belief in REA principles, noting that REA not only seamlessly produces all of the traditional accounting reports and financial statements, but also that it affords its users much, much more in the way of useful, non-account-based information. There are also other firms and standards bodies that champion REA principles as a set of messaging principles for inter-enterprise commercial exchange (International Standards Organization, 2015) and those standard bodies are presently undertaking new REA-based work in the area of blockchain technologies. Finally, there are practice groups that champion REA use in the development of ***open value networks*** -- peer-to-peer and open-collaboration collectives that operate on shared governance and value-creation principles that differ substantially from those of traditional business firms. This burgeoning REA practice may be seen as similar to the work on certain software systems whose theory preceded many successful implementations (like knowledge-based systems and relational database systems).
b. **Teaching**. In its working paper format, the REA manuscript had initiated a radical change in the way accounting information systems (AIS) were taught at both the undergraduate and graduate levels at Michigan State University in the late 1970s and early 1980s. MSU developed both network and relational accounting systems on mainframe platforms and used them extensively in teaching. The advent of the personal-computer (PC) age in the 1980s accelerated this instructional usage with the advent of PC-based relational systems, and these eventually began to stimulate the teaching of REA-based AIS at other colleges and universities. REA educational use was also accelerated in the 1990's by an increased emphasis of the fundamental importance of business processes in AIS teaching (McCarthy W. E., 2003a). In 2019, REA is the unifying focus in multiple AIS textbooks and a major course component in many others.
c. **Research**. Development of additional REA model components and methods involves explicitly normative research efforts conducted with design science (Hevner, March, Park, & Ram, 2004) methodologies and motives. As noted by McCarthy (2012), this field of inquiry has developed much more slowly in accounting than positivist approaches. Nonetheless, the 1982 model components have been extended in multiple ways in the REA literature (Dunn & McCarthy, 1997), in many cases aided by a close partnership among academics, practitioners, and standards bodies. There

> are also numerous empirical studies of REA use, and two recent research framework papers (Dunn, Gerard, & Grabski, 2016; Geerts, Graham, Mauldin, McCarthy, & Richardson, 2013) suggest avenues for further research development in both normative and positive directions.

Interoperability – again the key vision for REA – has also changed since 1982. The original REA paper invoked the interoperability vision of ANSI/SPARC (Tsichritzis & Klug, The ANSISPARC/X3/ DBMS Framework: Report of the Study Group on Database Management Systems, 1978) with a *conceptual schema* from which local views of data were derived with navigational and specificational programming. In computer science during the 1990s and early 2000s, that notion of a conceptual schema was augmented by the stronger semantic idea of ***ontologies*** (Sowa, 2000, p. chap. 2). Rzhetsky and Evans explain the nature and significance of that transformation (2011, p. 1):

> Historically, ontology was defined as philosophical inquiry into the nature and categories of existence. At the turn of the 20th century, logicians extended and formalized the notion of ontology as a system for describing entities that exist in the world (Luschei, 1962), their properties, interrelations, and inferential mechanisms for reasoning about them. In the 1990s, computer scientists reinvigorated and popularized the term by applying it to a wide range of machine-readable knowledge representations. Ontologies could be reused and shared as information schemas (Gruber, 1993). With the rise of scientific databases that are increasingly complex and persistent and require interoperability, ontologies have become enlisted in information technology used by many thousands of specialists worldwide.

Here, an ontology will be defined most simply as a formal representation of the categories within a domain. This includes an integrated set of concepts, commonly expressed as a semantic network of classes and relationships among those classes. Most importantly, such a representation should have a defined implementation vision that will facilitate intensional reasoning (Geerts & McCarthy, 2000a) about the properties of that domain. An ontology that is well-developed and well-specified serves as a starting point for the full categorical definition of a field.

The derivation, use, and aim of ontologies are summarized in the Ontolog Forum's Ontology Summit 2011 Communique (2011, p. 9):

> Fundamentally, ontology is about reaching agreements on what things mean and putting it in a machine-processible form. In an enterprise, this represents a radically different way to express meaning. The usual way is for meaning to be scattered randomly throughout the organization in people's heads, in email, in no-longer-maintained requirements documents, in conceptual models etc. In computational artifacts, a lot of meaning is in the names used to refer to things: code, variables, data base schema. Ontology both forces and enables an organization to be clear about what things mean and in doing so, gets everyone on the same page. Equally importantly, formally representing meaning enables automated inference which helps reduce unnecessary complexity, improve reliability, and increase agility.

REA modeling began to move toward an ontological basis in the late 1990s, and this movement resulted in its adoption by the International Standards Organization (ISO) as an interoperability standard – the accounting and economic ontology – for e-commerce messaging between trading partners. (International Standards Organization, 2007). It is extremely important to emphasize that in contrast to efforts in other areas of knowledge representation, the goal of ontology development in computer science is to serve as a basis for computable implementations. The REA ontology is certainly developed in this direction and with this emphasis.

Our goal in this monograph is to explain and to extend the elements of that ontology. Our exposition will be broken down into six chapters.

1. **Original REA model** -- We will introduce the original components of the REA accounting model as it was specified in the 1982 *Accounting Review* paper. Each of the basic classes will be reviewed, and changes in model components in use will be clarified and explained. These explanations will be frame-based, relying on class diagrams -- an advanced conceptual modeling notation that will be used to illustrate a simple purchasing example. This simple example will provide a gentle introduction to semantic modeling ideas for accountants, and conversely, a gentle introduction to accounting ideas for computer scientists.
2. **REA expansion in the granularity plane** -- The original REA model was specified as a business process (BP) modeling template where two economic agents were involved in requited transfers of economic resources with each other. In chapter 2, we will explain how those business process models can be aggregated into enterprise-wide value chains (Porter, 1985) and then further aggregated into market-wide value networks. We will also illustrate the decomposition of BP economic events into detailed workflow specifications, using the business process modeling notation (BPMN) of the Object Management Group (OMG, 2011). With the concept of a web of value creation activities fully explicated at four levels of aggregation, we will proceed to concentrate on REA temporal expansions at the business process level.
3. **REA expansion in the temporal plane** – Originally in 1982, REA modeled "what has occurred" in a business process – a traditional accounting emphasis that we now call the *accountability layer* of REA. We illustrate in this chapter two temporal expansions developed by Geerts and McCarthy (2002; 2006):

    a. the *policy layer* -- "what could be or should be" and,
    b. the *scheduling layer* -- "what has been specified or reserved."

    We will also explain here the nature of REA contracts and the extension of monitored commitments to accommodate value creation activities with dependent demand along a supply chain.
4. **REA business process extensions** – Having framed the context for REA business process definition in our first three chapters, we proceed here to discuss developed extensions for that model.

    a. contraction of the REA frame or pattern with the notion of "conceptual congruency" and expansion of that pattern with meronymic (part-whole relationships) development,
    b. accommodation of the two basic Coase (1937) process prototypes: market exchanges and within-firm conversions, and
    c. development of a 5-phase model for business processes: planning, identification, negotiation, actualization, and post-actualization. This extension was originally pioneered in ISO 15944-1 (2001) .

These three extensions will be illustrated with the construction and explanation of two REA process examples: a revenue cycle and a manufacturing (conversion) cycle.

5. **<u>The reorientation of REA components from an independent perspective</u>** – In McCarthy (2000) and in ISO (2007), the need for a view of economic phenomena from an independent or external perspective was recognized. The ***Independent view*** augments and extends the conventional ***Trading-partner*** or internal view long ascendant in traditional accounting. We explain here how this new view extends some basic REA definitions and how those new definitions can be viewed in the more traditional trading partner perspective. We will also explain here the burgeoning phenomena of open value networks as they are specified with Independent view REA principles and how those principles relate to another new technology: the use of blockchains or distributed business transaction repositories.
6. **<u>Necessary extensions and future directions in REA work.</u>** – Many parts of the REA ontology are still under-specified and under-developed. The need for REA extensions has long been recognized, but their detailed development is still an open work item within the semantic modeling of accounting phenomena research community. These under-developed items include (1) the use of business process state machines, (2) the expansion of automated reasoning capabilities within REA, (3) the specification of claims (debt and equity) and their alignment with other financial ontologies, (4) the expanded components of a procedure hierarchy designed to materialize general ledgers for financial accounting reporting purposes, (5) the accommodation within REA structures of newer conceptual methods for advanced types of costing championed by the Institute of Management Accountants (IMA) (2012), and (6) reconciliation of REA participation and control principles with accepted best practices in the areas of corporate governance and control frameworks. Our explanations in this chapter will concentrate on areas where the monograph authors have already done extensive research, but their framing we hope will encourage additional research by others.

Again, the key to understanding various REA components is the model's emphasis on interoperability or, as stated more formally, on ***ontological commitment***. We intend that this monograph will provide enough of a review of REA ideas from many different sources that potential researchers, teachers, or practitioners can understand and adopt those features to their own particular use. On the first pass through, we intend that there will be little extended need for readers to consult the original journal, presentation, and standard sources. For more detailed understanding and serious proposal of research extensions, we envision that this monograph will provide a convenient one-stop starting point.

William E. McCarthy, Guido L. Geerts, and Graham Gal
East Lansing, Michigan
February 2019

Chapter 1: The Basic Resource-Event-Agent (REA) Model

A Simple Example Modeled with Two Different Kinds of Accounting Systems

An accounting system is a transactional repository for two types of economic transactions:

(1) between-company market *transfers* that exchange one set of resources for another (like buying eggs and flour for cash in a market exchange) and
(2) within-company *transformations* that convert one set of resources into another set (like using eggs and flour to make cookies in a manufacturing process).

Most commonly in introductory accounting instruction, an accounting system is conceptualized as a series of entries (reflecting those transfers or transformations) into various accounts, following the mechanisms of double-entry bookkeeping. These mechanisms post monetary amounts to either the *debit* side of an account (usually the left) or the *credit* side (usually the right) in pair-wise fashion based on a well-defined set of conventions. Computerized general ledger systems use these concepts in a way very similar to those manual bookkeeping methods.

The Resource-Event-Agent (REA) accounting model (McCarthy W. E., 1982) is the subject of this monograph, and we begin this first chapter with an example that illustrates its most basic features. REA accounting systems take an approach different from double-entry bookkeeping in that they use a modeling pattern to develop a database for the same purpose of producing a transactional memory of between-company resource transfers and within-company resource transformations. REA does not use accounts as modeling primitives, nor does it follow debit-credit posting procedures. Instead, it builds a linked database where rows in different tables are connected to capture the semantics or meaning of the modeled transactions.

In the introductory section of this first chapter, we will look at a simple example to see how both these types of accounting systems work. Our purpose in such an exposition is **not** to compare the features/benefits of each system (that will definitely come later in the monograph). Rather we intend to use the familiar ideas of double-entry bookkeeping that most of our intended primary audience (thought to be accounting researchers, educators, and practitioners) know quite well to show how an REA accounting system might work. Conversely, this comparative example might additionally appeal to a secondary group of computer-science-oriented readers who can use their familiarity with relational database systems to gain insight into double-entry conventions and the connection of those conventions to REA modeling.

After these simple examples have been explained, we will proceed to a foundational description of the basic REA model, upon which the remaining chapters of the monograph will depend.

A Simple Set of Transactions Modeled With Double-Entry Accounts

In Figure 1-1, we list an introductory description of actions to create the company, followed by eight transactions that detail (in natural language terms) how a cookie-making company might come to acquire its initial set of raw materials for eventual production.

Illustrating how that simple transaction set might be recorded in both double-entry terms and in REA terms is shown in the text and figures that follow.

0. On June 30th, Judy Wilson, company owner, contributed $10,000.00 to Wilson Company. Later that day, the company spent $1,000.00 on building rental and $1,000.00 on a machine acquisition (a baking oven). The company also hired a number of employees to serve as buyers, cashiers, salespeople, drivers, and manufacturing workers.

---

1. On July 1st at 09:00, Alice purchased some eggs (8 dozen @ 5.25) and butter (2 packages @ 3.13) from Allen Company with a total cost of $48.26.
2. On July 1st at 09:30, Mary disbursed $30.00 to Allen company in partial payment for transaction#1.
3. On July 1st at 11:00, Ted purchased some sugar (20 bags @ 12.10) and flour (9 bags @ 5.30) from Smith Company with a total cost of $289.70.
4. On July 2nd at 09:00, Bob purchased some walnuts (4 bags @ 7.90) from Smith Company with a total cost of $31.60.
5. On July 2nd at 12:00, Mary disbursed $18.26 to Allen company to complete payment for transaction#1.
6. On July 3rd at 08:00, Chris disbursed $321.30 to Smith company in full payment for transaction#3 and transaction#4.
7. On July 3rd at 09:30, Alice purchased some eggs (4 dozen @ 5.15) from Jones Company with a total purchase cost of $20.60.
8. On July 5th at 08:00, Carol disbursed $20.60 to Jones company in full payment for transaction#7.

Figure 1-1 – Sample Transactions for Acquisitions Cycle

Figure 1-2 illustrates how the sample transactions might be booked in double-entry T-accounts. For example, transaction #1 portrays Alice's July 1st purchase of eggs and butter as a 48.26 debit (left side) to the raw-materials account and a 48.26 credit (right side) to the accounts-payable account. Similar treatment is accorded her purchase of eggs on July 3rd (transaction #7) -- a debit to raw-materials of 20.60 and a credit to accounts-payable for 20.60. By contrast, transaction #8 is actually the requited payment for transaction #7 with a debit to accounts-payable for 20.60 and a credit to the cash account for 20.60.

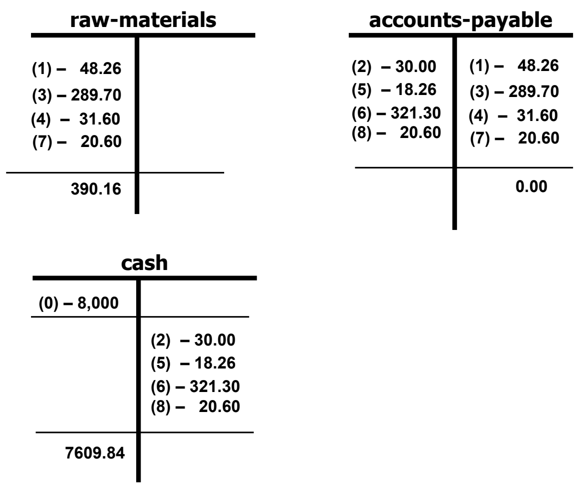


Figure 1-2 – T-accounts for Sample Acquisition Transactions

Where a line is drawn across an account on both sides, this signifies that the account has been totaled over some period of time. For example, the cash account starts from a balance of $8,000.00 at the beginning of July, resulting from the June 30th transactions (10,000.00 minus the two 1,000.00 expenditures on building rental and supply purchases).[2] The cash account has also been totaled after transaction #8, leaving a debit balance of $7609.84.

In modeling an entire enterprise with cycles for financing, payroll, manufacturing, and revenue, this small set of accounts would be enlarged significantly. However, its basic recording, aggregating, and reporting mechanisms (and their accompanying procedures) would remain essentially the same.

A Simple Set of Transactions Modeled with an REA Accounting System

By contrast with the example of Figure 1-2, Figures 1-3(a) and 1-3(b) illustrate how the eight example acquisitions transactions might be modeled in a linked relational database (Codd, 1970) based on the REA model. Each table whose name begins with a capital letter (like Purchase, Vendor, or Buyer) represents a class or representational category. These class tables have an identified "key" attribute (underlined) whose instances identify a particular row in the table. For example, we have three vendors, represented by key values of "V-888," "V-987," and "V-988."

Purchase

Purchase Timestamp	Dollar Amount
1JUL0900	48.26
1JUL1100	289.70
2JUL0900	31.60
3JUL0930	20.60
---	---

lineItem

Purchase Timestamp	Product Number	Quantity	Unit Price
1JUL0900	P-2	8	5.25
1JUL0900	P-5	2	3.13
1JUL1100	P-3	20	12.10
1JUL1100	P-1	9	5.30
2JUL0900	P-4	4	7.90
3JUL0930	P-2	4	5.15
---	---	---	---

RawMaterial

Product Number	Description	Unit	QOH
P-1	flour	bag	9
P-2	eggs	dozen	12
P-3	sugar	bag	20
P-4	walnuts	bag	4
P-5	butter	package	2
---	---	---	---

partyTo2

Purchase Timestamp	Vendor Number
1JUL0900	V-888
1JUL1100	V-987
2JUL0900	V-987
3JUL0930	V-988
---	---

Vendor

Vendor Number	Vendor Name	Present Accounts Payable
V-888	Allen Company	0.00
V-987	Smith Company	0.00
V-988	Jones Company	0.00
---		

partyTo1

Purchase Timestamp	Buyer Employee Number
1JUL0900	E-1234
1JUL1100	E-1235
2JUL0900	E-1236
3JUL0930	E-1234
---	---

Buyer

Employee Number	Buyer Name	Buyer Rating
E-1234	Alice	R-9
E-1235	Ted	R-10
E-1236	Bob	R-9
---	--	---

Figure 1-3 (a) – Partial Database for Acquisitions Cycle

[2] For simplicity, three other accounts for the June 30th transactions -- equity with a credit balance of 10,000, rental expense with a debit balance of 1,000.00, and oven equipment with a debit balance of 1,000 have been left off the acquisition example.

paysFor

Purchase Timestamp	Disburse Timestamp	Amount Applied
1JUL0900	1JUL0930	30.00
1JUL0900	2JUL1200	18.26
1JUL1100	3JUL0800	289.70
2JUL0900	3JUL0800	31.60
3JUL0930	5JUL0800	20.60
---	---	---

CashDisbursement

Disburse Timestamp	Amount
1JUL0930	30.00
2JUL1200	18.26
3JUL0800	321.30
5JUL0800	20.60
---	---

paidFrom

Disburse Timestamp	Account Number
1JUL0930	A-2
2JUL1200	A-2
3JUL0800	A-2
5JUL0800	A-2
---	---

Cash

Account Number	Present Balance
A-1	3000.00
A-2	4609.84

partyTo3

Disburse Timestamp	Vendor Number
1JUL0930	V-888
2JUL1200	V-888
3JUL0800	V-987
5JUL0800	V-988
---	---

partyTo4

Disburse Timestamp	Cashier Employee Number
1JUL0930	E-1220
2JUL1200	E-1220
3JUL0800	E-1222
5JUL0800	E-1221
---	---

Cashier

Employee Number	Cashier Name	Bonding Status
E-1220	Mary	true
E-1221	Carol	false
E-1222	Chris	true
---		---

Figure 1-3(b) – Partial Database for Acquisitions Cycle

Each table of Figures 1-3(a) and 1-3(b) whose name begins with a small letter (like lineItem and partyTo2) represents a connection between representational categories. For example, the partyTo2 table shows the association between Vendors and Purchases, and the paysFor table shows the association between Cash Disbursements and Purchases. At the instance level of the first two rows of the lineItem table, we can see connections between the single purchase of Transaction #1 (key value of "1JUL0900") and two instances of RawMaterial (key values of "P-2" and "P-5").

By following keyed entries in the various tables, readers can see how all of the Figure 1-1 transactions are modeled, but we highlight here the treatment of transaction #7 and transaction #8. We explain these below, but for ease of use, we have highlighted transaction #7 with solid black ellipses and transaction #8 with dotted black ellipses.

- From tables shown in Figure 1-3(a), we can follow the purchase of 3 July at 09:30 as consisting of 4 dozen eggs (product "P-2") at a unit price of 5.15. The parties to this transaction were the vendor "Jones Company (V-988)" and the buyer "Alice (E-1234)."
- From the tables shown in Figure 1-3(b), we can see that the purchase of "3JUL0930" was paid for with the cash disbursement of "5JUL0800". This disbursement came from cash account "A-2," and the parties to this transaction were cashier "Carol" and the vendor "Jones Company."

By following the key values in the two instances above, we can see the storytelling capabilities of the linked relational tables of Figure 1-3. Such multidimensional and disaggregate representation is a key feature of REA accounting systems. Of course in an actual computerized implementation, the linking of various rows in separate tables would be done programmatically rather than manually.

A Semantic Model of the Relational Tables

In Figure 1-4, we move up a level of abstraction with a drawing that shows the categories of Figure 1-3 (tables beginning with capital letters) as rectangles being connected by named arcs (line labels beginning with small letters).[3] Figure 1-4 is actually a map of Figure 1-3, with most of the top part deriving from the part (a) tables and most of the bottom part deriving from the part (b) tables.

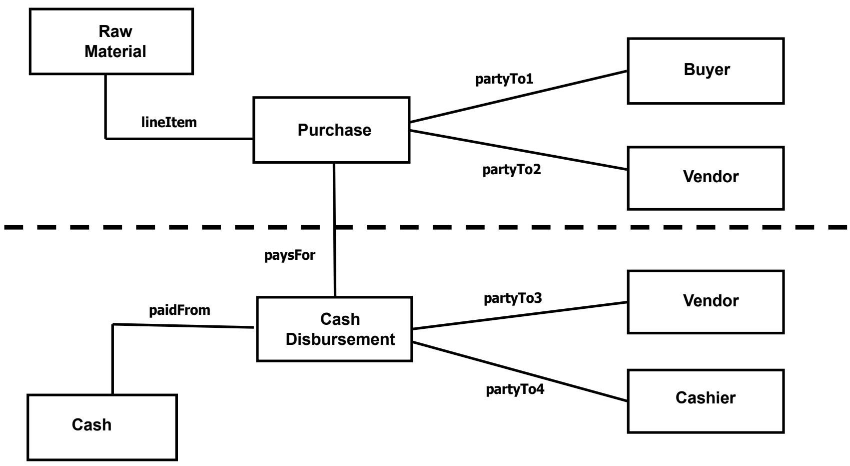


FIGURE 1-4 – Class Diagram for Acquisition Business Process

More formally, Figure 1-4 is a *UML class diagram*[4] that shows the classes (fundamental categories) of a particular domain example with termed links (associations between those categories). Deriving Figure 1-4 from the raw transaction descriptions of Figure 1-1 reflects semantic modeling work in accounting initially accomplished in McCarthy (1979).

In Figure 1-5(a), we follow McCarthy (1982) by moving up another level of abstraction, generalizing some of the categories of Figure 1-4. For example:

- "Cash" and "Raw Material" are seen as subtypes of "Economic Resource;"
- "Purchase" and "Cash Disbursement" are seen as subtypes of "Economic Event;" and
- "Buyer," "Vendor," and "Cashier" are seen as subtypes of "Economic Agent."

[3] The "Vendor" class is illustrated twice for symmetry reasons.
[4] UML stands for Unified Modeling Language (Object Managment Group, 2012). UML artifacts portray both structure and behavior. The class diagrams used here depict the semantic structure of a model.

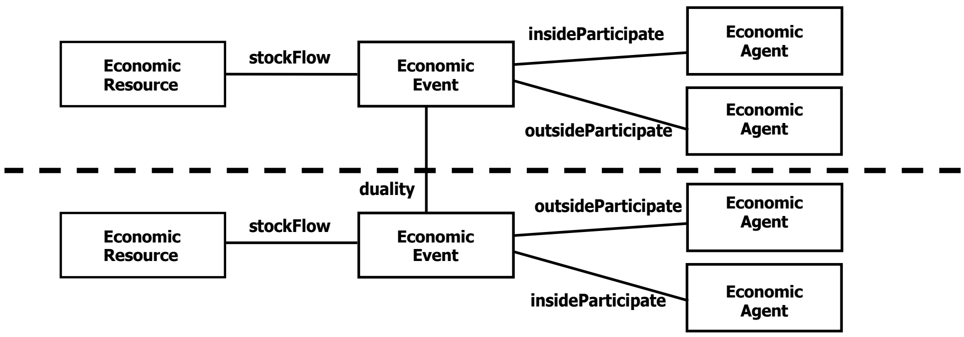


Figure 1-5(a) – The REA Metamodel: A Generalized Class Diagram for a Business Process

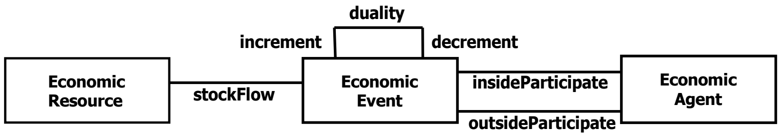


Figure 1-5(b) – The REA Metamodel with Symmetrical Halves Folded Together

The associations (named arcs) are also generalized, and the result is actually a template for modeling business processes or accounting cycles in general.

In Figure 1-5(b), we fold the two halves of 1-5(a) on top of each other to illustrate the REA metamodel.[5] The left to right alignment of the economic categories (Resource-Event-Agent) is what gave the REA model its original name (McCarthy 1982).

The REA Metamodel at Three Levels of Abstraction

In knowledge representation terms, REA is a *data model pattern* for economic exchanges or conversions. Our acquisition process example (Figure 1-1) is an illustration of REA semantics expressed in natural language at three levels of increasing generality:

- ***Level-0*** *Alice purchased four dozen eggs from Jones Company on the 3rd of July, and subsequently on the 5th of July, Carol disbursed $20.60 for that purchase to Jones Company* (the marked rows of Figure 1-3).
- ***Level-1*** *Buyers purchase raw material from vendors, and in return, cashiers make cash payments to those vendors* (Figure 1-4).
- ***Level-2*** *There is a transaction (an economic event) where an inside agent (like an employee who works for the firm) receives something of value (an economic resource) from an outside agent (an independent party external to the firm); this increment event is paired with a mirror-image decrement event where the inside agent gives in kind another economic resource to the outside agent (Geerts and McCarthy 2002). The presumption is that the received resource has more value to the enterprise in its pursuit of its entrepreneurial goals* (Figure 1-5).

[5] The "duality" association in Figure 1-5(b) is a link from one subtype of Economic Event to another subtype of Economic Event. Such a self-referent link is called a *recursive association* in modeling terms. Recursive associations in UML are often augmented with role names, as is done here with the labels of "increment" and "decrement" in Figure 1-5(b).

Our names for the increasing levels of generality are adapted from the Meta-Object Facility (MOF) of the Object Management Group (2014). MOF Level-0 represents an actual **instantiation** of an REA process, MOF Level-1 represents an **application** of REA principles to an actual accounting cycle, and MOF Level-2 is a natural language description of the REA **metamodel** of Figure 1-5.

In Figure 1-6, we reverse our generality direction and outline how these different MOF levels of the REA pattern are portrayed as UML artifacts in the top-down fashion with which they are normally discussed.

- The REA metamodel at Level-2 is illustrated here as a simple UML class diagram with the names of the REA categories being shown as classes and their links being illustrated with named associations.
- An REA application at Level-1 is illustrated as a UML class diagram, sometimes with a single partition of the class box, but more commonly with a second partition showing ***attributes***. In some cases, a third partition might illustrate methods. Above the class and association names, their ***stereotypes*** are illustrated within double brackets. Readers should note that the use of stereotypes at Level-1 provides an easy link to metadata at Level-2.
- An REA instantiation at Level-0 is illustrated as an object diagram with instances underlined per UML convention. Here as well, readers should note that the use of **classifiers** at Level-0 provides an easy link to Level-1 class diagrams. UML object diagrams provide an implementation-free vehicle to illustrate instance data, and with the exception of posted links, object diagrams correspond quite closely to the rows in a relational database. By convention in UML, an object or instance is represented with a simple article in front of its class name (for example, the object “aVendor” is an instance of the class “Vendor”).

Figure 1-6 illustrates modeling notation for REA at three MOF levels, and Figure 1-7 continues by showing that notation for part of our simple acquisition example. Figure 1-7 is actually a reconfiguration of the bottom parts of Figure 1-4 and Figure 1-5(a) together with the instance data for transaction #8 (dotted ellipsis) of Figure 1-3(b) in UML. Readers should note how stereotypes at Level-M1 and classifiers at Level-M0 in Figure 1-7 allow the three levels to be integrated.

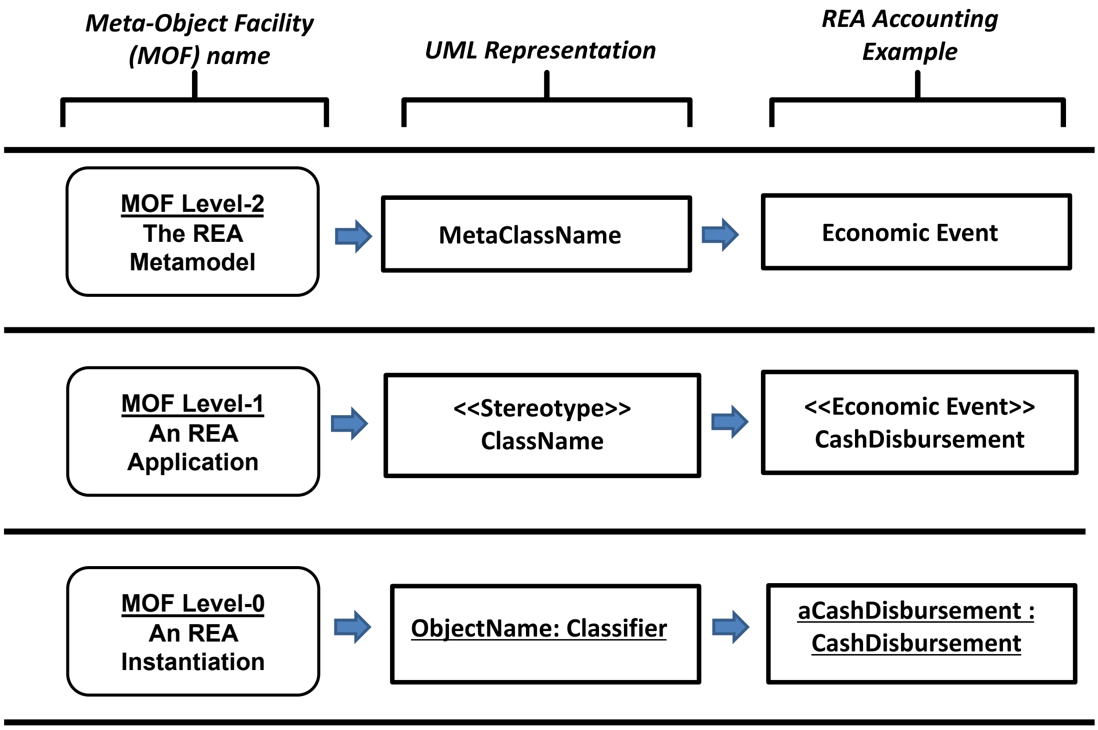


Figure 1-6 – Shorthand Use for the Meta-Object Facility (MOF)

Additionally in Figure 1-7, we have added two more components of UML diagrams that augment the information shown in a comprehensive fashion.

- **Attributes** of classes which are illustrated within the confines of the second partition of each category in Figure 1-7(b). For example, the "Cash" class has attributes of "accountNumber"and "presentBalance," with the former serving as the primary key (PK) of the class (that is, its identifying characteristic).
- **Multiplicities** of associations which are illustrated as "minimum..maximum" pairs on each line. Minimum multiplicity values may be either **0** (optional) or **1** (required), while maximum multiplicity values may be either **0** (no instances) or * (many instances).[6] For example, in Figure 1-7(b):
  - the "1..1" notation to the right of Cash indicates that a single cash disbursement must relate to some cash account minimally, and that it also can relate to just one account maximally;
  - the "0..*" notation to the left of CashDisbursement indicates that a single cash account may exist without any connections to cash disbursements minimally, and that it may also relate maximally to many cash disbursements;
  - the "0..1" notation to the left of Vendor indicates that a single cash disbursement may exist minimally without any connections to vendors (for example, payments to employees for payroll purposes), but that maximally a single payment can be applied to just one vendor if that is what the payment is used for.
  - The "0..*" notation to the top right of Cash Disbursement indicates that a vendor may exist before any disbursements are sent to that company

[6] A full UML model allows for more comprehensive values for minimum and maximum multiplicities. For simplicity, we have limited the values for these multiplicities.

(minimum), but that over the course of time, many payments may be sent to it (maximum).

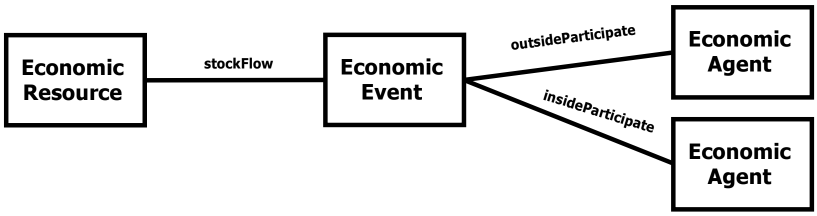


Figure 1-7(a) – REA Pattern at Level-M2 – the REA Metamodel (partial)

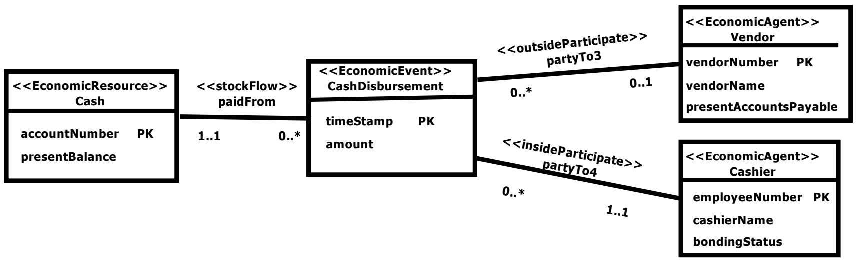


Figure 1-7(b) – REA Pattern at Level-M1 – An REA Application

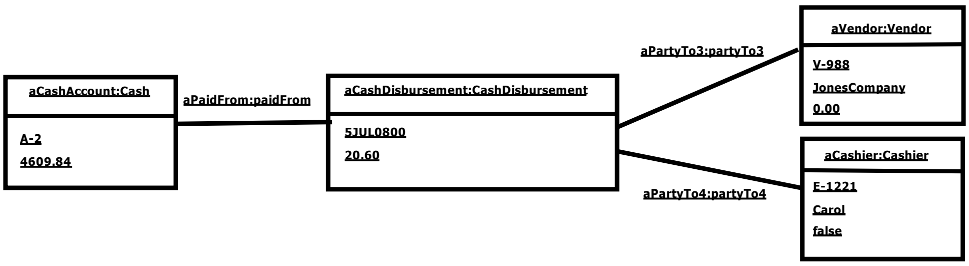


Figure 1-7(c) – REA Pattern at Level-M0 – An REA Instantiation

For comprehensiveness purposes, we illustrate the complete M1 representation of our acquisition example in Figure 1-8. Readers should note there that it is sometimes possible for associations to have separate attributes of their own (for example in the "lineItem" and "paysFor" associations). In these cases, the attributes are shown in a second box partition, which is then connected to the association link with a dotted line. Such attribute-bearing associations are called **association classes** in UML.

Figure 1-8 completes our exposition of the initial acquisition example first enumerated in Figure 1-1. This particular example and its notation will be used extensively throughout our succeeding chapters of the monograph.[7]

[7] This acquisition example corresponds very closely to the first simple example of REA used in the original REA exposition (McCarthy 1982 p. 566). Readers interested in REA changes accumulating in use over 30 years may compare the small differences in notation and structure.

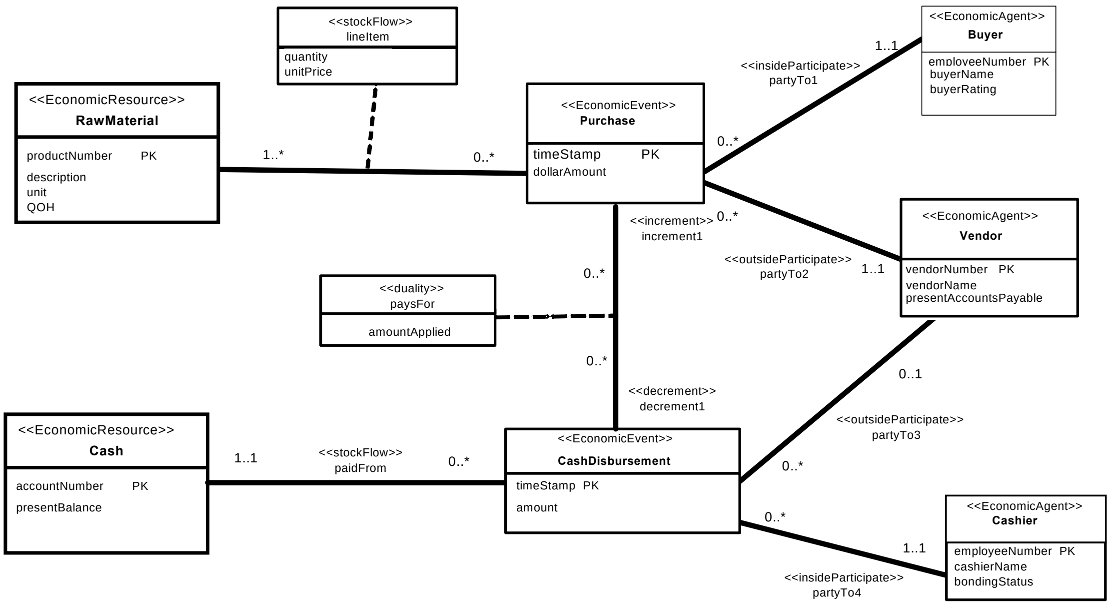


FIGURE 1-8 – M1 Class Diagram for Acquisition Process (an REA Application) with Attributes, Multiplicities, and Stereotypes

Expanding and Redefining the Concept of Economic Agent

In the original REA paper, the concept of Economic Agent had two subsets -- agents inside the enterprise, and agents outside of the enterprise -- with Economic Event requiring the participation of both in a ternary (3-way) relationship. We have simplified that participation in our metamodel of Figure 1-5, replacing the ternary connection with two binary associations labeled "***insideParticipate***" and "***outsideParticipate***". This decomposition simplifies the metamodel and allows some extensions to be made later to the inside-outside distinction when we introduce the notion of independent view modeling in Chapter 5.[8]

Additionally, in the original REA paper, Economic Unit was a subset of Economic Agent that was congruent with inside agents. However, the diagrammatic designation of the Economic Unit entity there did not reflect the full complexity of the accompanying text, so we intend here to clarify and extend those meanings in UML format.

The generalization hierarchy for our updated view of agents is shown in Figure 1-9. The top of that figure illustrates the most generalized case of "Economic Agent" with a recursive "responsible" association having roles of "superior" and "subordinate". The generalization arrow indicates a subset or an "is-a" relationship in UML. The second hierarchical level is both complete and disjoint, this means that the set union of the second level classes ("Economic Unit" and "Person") is congruent with "Economic Agent." In non-set-oriented terms, this means that every economic agent is either an economic unit or a person. More specifically for our second level classes, we have these clarifications.

[8] As noted by Weber (1986), the ternary link might, in very rare cases, be subject to multivalued dependency anomalies. So the simpler binary use is preferred for that reason as well.

- Human economic agents are classified as *Person* -- a term instituted and used extensively in ISO 15944-1 (2001), although we are restricting its coverage somewhat in comparison to that standard. Persons are human agents who can enter into economic exchanges and conversions and who can make commitments to execute future transactions.
- The concept of *Economic Unit* is now refined to designate non-human agents such as companies, divisions, and departments.
- Finally, an assign association is used to pair a Person to an Economic Unit.

Readers should note that these agent decompositions in a generalization hierarchy are subject to needed use (Denna, Cherrington, Andros, & Hollander, 1993); that is, a modeler only decomposes to the level needed for planning, controlling, and evaluating the object reality. In most cases, it will suffice to simply use the hierarchy's top level – Economic Agent – to explain the modeling of accounting and economic reality with particular sets of simple examples. In fact, we will be using that method throughout the rest of the monograph. However, for readers interested in the extended use of the agent generalization hierarchy of Figure 1-9, we provide a detailed example (at MOF Level-1) in an appendix to this chapter.

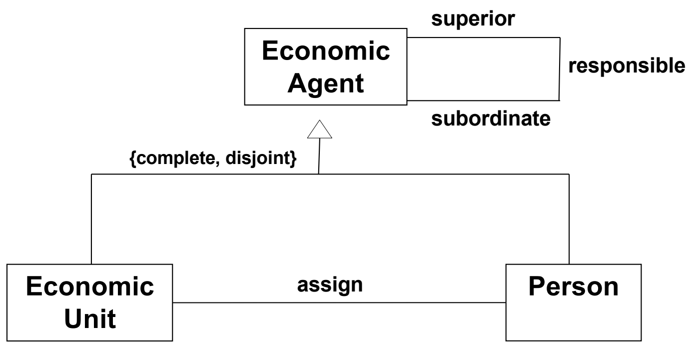


Figure 1-9 – Agent Generalization Hierarchy

The Basic Elements of the REA Ontology

It is our prime purpose in this monograph to explain REA at the metamodel level, where each class and association will be given a specific definition. In our preface, we equated the REA metamodel with the computer science concept of a ***domain ontology*** which (again) most simply is a representation of a set of categories within a domain and the relationships that connect those concepts. For us, the REA metamodel at MOF Level-M2 is a domain ontology for accounting.

When the agent extensions are added to the previously derived metamodel of Figure 1-5, a new operational REA metamodel (at Level M2) emerges, as seen in Figure 1-10. Here are the basic definitions we intend to use for each of the metamodel components, as we speak of them from the perspective of a central object enterprise. Readers will notice that most of the substance of these definitions comes from the original REA paper (McCarthy W. E., 1982)

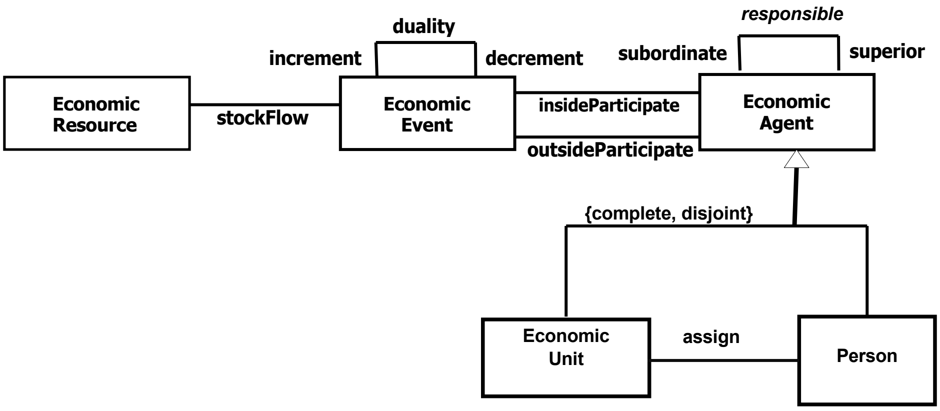


Figure 1-10 – The REA Metamodel with Agent Generalization

Classes:

- ***Economic Resources*** are groups of objects that (1) are scarce and have utility and (2) are under the control of an economic agent. Economic Resources may be subtyped into goods, services, rights, or a combination of these three (ISO 2007). Such a combination portrays resources as a portfolio of attributes of value to an economic agent (Lancaster 1966). This definition of a resource is adapted with slight modification from the wording of McCarthy (1982, 562), Ijiri (1975, pp. 51-52), and Lancaster (1966).
- ***Economic Events*** are classes "of phenomena which reflect changes in scarce means [economic resources] resulting from production, exchange, consumption, and distribution" (Yu, 1976, p. 256). This definition is unchanged from the original McCarthy (1982, 256) paper.
- ***Economic Agents*** are identifiable parties (1) with discretionary power to use or dispose of economic resources, or (2) who are responsible for subordinates' use or disposition. Again, this definition is adapted with slight modification from the wording of McCarthy (1982, 563) and Ijiri (1975, 51-2). Economic Agents have these subtypes:
    - *Persons* are human agents who can enter into economic exchanges and conversions and who can make commitments to execute future economic events, and
    - *Economic Units* are the non-human subset of economic agents, such as companies, divisions, and departments.

Associations:

- ***stockFlow*** associations link a continuant – Economic Resource – to an occurrent – Economic Event -- in a relationship that maintains data consistency, effected with either triggered or adjusting procedures. (Geerts and McCarthy 2002, McCarthy

1982).[9] As McCarthy (1982, p. 562) notes "…a perfectly consistent [database] schema would require both a new instance of this relationship type and a new update or instance of a resource entity type for every new event entity". In the ***Trading Partner view*** (ISO 2007)[10] of the REA model, stockFlow associations may be decomposed into ***inflow*** and ***outflow*** when considered from the relative perspective of the object enterprise being modeled. Additionally (as illustrated on the left of Figure 1-11), ***outflow*** may be further decomposed into ***give*** for REA market exchanges and ***consume*** or ***use*** for REA internal conversions.[11] Similarly (as illustrated on the right of Figure 1-11), ***inflow*** may be decomposed into ***take*** for exchanges and ***produce*** for conversions.

- ***duality*** associations link requited Economic Events where one occurrent flow is the economic or legal consideration given/obtained for the other (Mattessich, 1964). Again, in the trading partner view of REA, duality is shown as a recursive association in Figure 1-10 where one occurrent (or occurrent set) plays the role of a resource ***increment***, while the other occurrent or set plays the role of a resource ***decrement.***
- ***participate*** associations link multiple Economic Agents to an occurrence of an Economic Event. As illustrated again from the perspective of a central object enterprise in Figure 1-10, these agents represent the competing economic interests of a party inside the firm (insideParticipate) and another one outside the firm (outsideParticipate).
- ***responsible*** associations establish hierarchies of economic agents with higher level agents (***superior*** role) being accountable for the activities of lower level agents (***subordinate*** role) (McCarthy 1982, p. 564). ***Responsible*** associations are used to reflect the organizational hierarchies that are both described extensively in the microeconomic literature (Kroszner & Putterman, 2009), and utilized extensively in the organizational coding schemes of many accounting/enterprise software packages.
- ***assign*** associations link a Person to an Economic Unit, reflecting the positioning of agents in an organizational hierarchy.

[9] Continuant and occurrent are class categories that link to an upper level ontology, most specifically that of John Sowa (2000). Expanding on Sowa, Geerts and McCarthy (2002, p. 2) explain these terms as follows:
… a Continuant is an enduring object that "has stable attributes that enable its various appearances at different times to be recognized as the same individual" while an Occurrent is a process or event that " is in a state of flux" and that "can only be identified by its location in some region of time-space" (Sowa 2000, p. 71).

[10] In approximate terms, the Trading Partner perspective of REA corresponds to its use in enterprise-centered information systems like ERP (Enterprise Resource Planning) systems. The Independent perspective of REA corresponds to its use in collaboration space information systems, such as proposed by the ISO (2007) open-EDI initiative. A prime example of such an information system structure in 2019 is a blockchain. The mapping and alignment of these two perspectives is the subject of Chapter 5 in the monograph.

[11] The difference between REA market exchanges and REA internal conversions is addressed in considerable detail in Chapter 4 of the monograph.

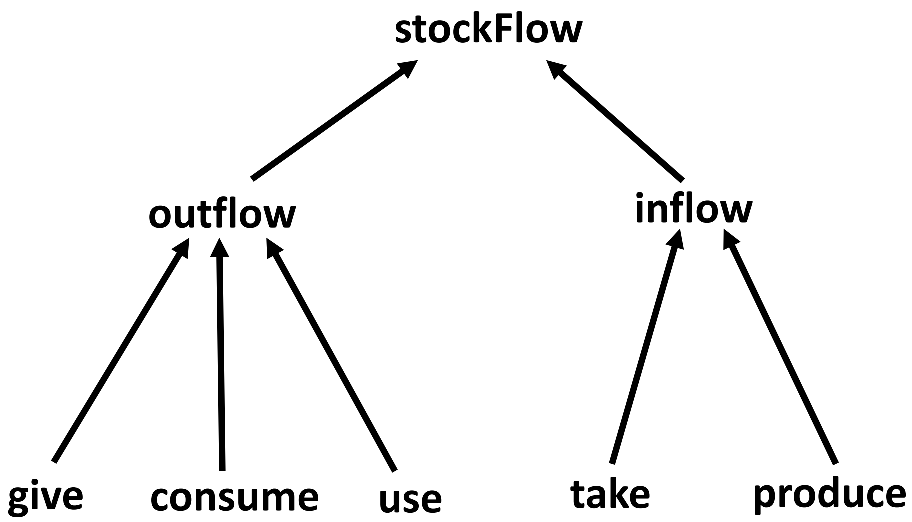


Figure 1-11 – Stockflow Decomposed

Informal Explanations of the REA Pattern

Before summarizing the Chapter 1 explanations of the REA metamodel, it is informative to view that pattern from some simpler perspectives than those provided by the formal UML artifacts that we have relied on so heavily in this initial chapter. The basic REA pattern is widely used in accounting instruction, a fact attributable in our estimation to its ability to explain intuitively the overall nature of business processes. Two such simplified examples follow, both of them adopted from teaching uses of REA.

A business process, most generally, is a set of activities that transforms a set of inputs into a more desired (or of higher utility in economic terms) set of outputs. A first pass explanation of a process might ask these simple questions:

- what resources were involved for input and output?
- when did the activities occur?
- who was involved?
- why were the activities completed?

As portrayed in Figure 1-12, the REA pattern provides slotted answers to these questions as a preliminary analysis tool. The decrement event shows *when* it occurred, *what* resource was transferred out, and *who* was involved, both inside and outside the firm. The increment event shows the same for what was acquired. The *why* question is answered by connecting decrements to increments via duality associations. In later chapters, we will extend this *why* rationale by showing how decrement-increment processes can be connected together via value chains. We will also explore later the *how* and *where* questions of business processes by adding in workflow specification and business location data.

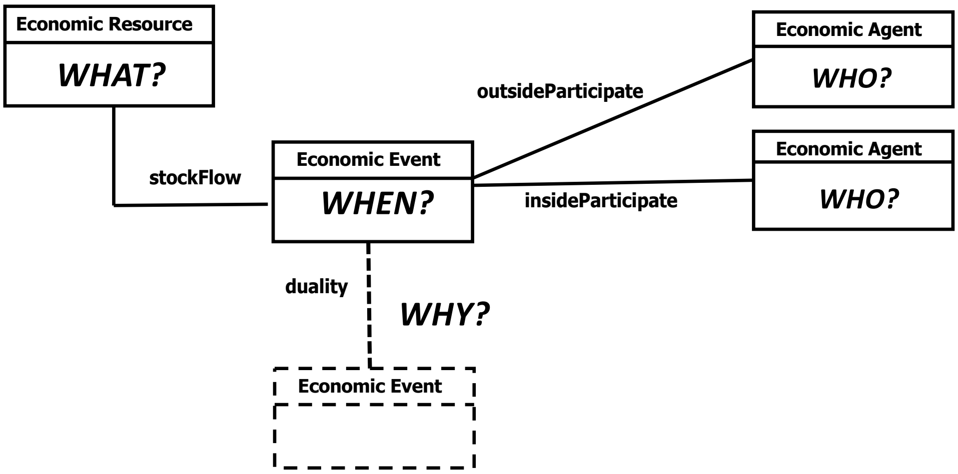


Figure 1-12 – The What, When, Who, and Why Translation of a Business Process

As a first step in understanding the semantics of natural language text like the transaction descriptions of Figure 1-1, some artificial intelligence (AI) systems like *Watson* (Ferrucci, et al., 2010) apply linguistic analysis to produce diagrammed sentences for understanding purposes. Such natural language parsing is based on case-grammars, a concept initially proposed by Fillmore (1967). In Figure 1-12, we illustrate in a very preliminary way such analysis with REA structures.

Figure 1-13(a) illustrates (at MOF Level-2) how an REA pattern can be used to translate a "subject--active verb--direct object--prepositional phrase" sentence or vice-versa (sentence to REA model). This is an important example of the trading partner view (ISO 2007) which was used in the original REA paper and which actually underlies the traditional accounting equation of "assets = liabilities + owners' equity". In this view, the inside agent becomes the subject for the active verb, because economic matters are seen from that agent's relative perspective. As we will see in Chapter 5, there is an alternative perspective -- called the independent view -- needed in open value networks where economic matters are seen from a more neutral perspective (ISO 2007).

Figure 1-13(b) portrays (at MOF Level-0) how the instances of transaction #7 and transaction #8 in Figure 1-1 are shown as a diagrammed compound sentence where Alice's purchase of eggs is compensated by Carol's disbursement to the same vendor from the trading partner perspective. Not shown here is another relative view from the perspective of the Jones Company which would actually be recorded (with the present technology of most accounting systems) in the database of Jones. There, "Jones Company" would be the subject of the two clauses and the active verbs would change to "sold" and "received," the REA events would change to "Sale" and "Cash Receipt," and Alice and Carol would be modeled as representatives of a "Customer" named Wilson Company. Again as we shall see in Chapter 5, the REA independent perspective (ISO 2007) will offer a solution to this redundant modeling.

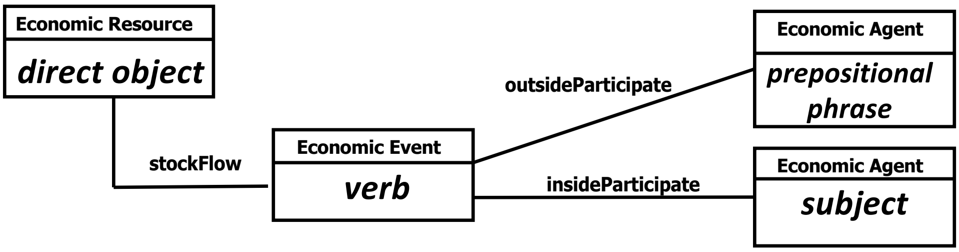


Figure 1-13(a) – Normal Sentence Translation of the REA Metamodel

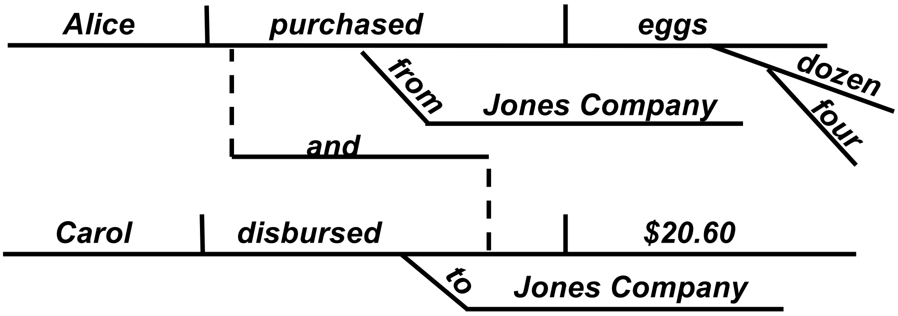


Figure 1-13 (b) – Transactions #7 and #8 Diagrammed as an REA-Patterned Compound Sentence

If we combine the ideas of Figure 1-12 and Figure 1-13, we could envision how an advanced query system (with a natural language interface) might work on an REA-patterned database with example questions like these:

- **Who** was responsible for the purchase of eggs on July 3rd at 09:30?
- **Why** did Carol send $20.60 to the Jones Company on July 5th at 08:00?
- **What** are the patterns of delivery used by our dairy (eggs and butter) vendors?

Certainly, the first two of these questions would be quite easy to construct. The third might need expansion of the attribute set used so far with more transaction data and more links, but those are exactly the kinds of extensions we will address in Chapters 3 and 4.

Chapter Summary

In McCarthy (2003a), an important quote from the French computer scientist Jean-Raymond Abrial was used to introduce the central idea of semantic database modeling, an idea that motivated the development and use of the original 1982 REA model:

> We shall define a database as the model of an evolving physical world (Abrial, 1974, p. 3).

McCarthy said (2003a, p. 428) that REA's core feature "was an object pattern consisting of two mirror-image constellations that represented semantically the input and output components of a business process". He then went on to repurpose and expand the Abrial quote for accounting:

> An accounting database is a model of the reality surrounding an evolving business enterprise, including its past set of accountability transactions, its present set of commitments and claims, and its future set of plans and policies (p. 428).

Chapter 1 has summarized and explained the symmetrical constellation of resources, events, and agents, both as those components were originally explained in 1982 and as they are used now. This represents the past set of accountability transactions referenced above. In Chapter 3, we will address the second and third part of that quote – the present set of commitments and claims and the future set of plans and policies. Before defining that expansion however, we move first to a redefinition and expansion of the basic idea of a "business process." In Chapter 2, we explain how REA-modeled business processes can be combined into enterprise-wide value chains and then further into economy-wide value networks. We also elaborate in Chapter 2 on the decomposition of business processes into workflow components of value-added activities.

Chapter 1 Appendix – Use of Agent Subtypes in Modeling Responsibility Charts

Figure 1-9 illustrates, in the generalization plane, the updated decomposition of the "Economic Agent" primitive of the REA model. This appendix shows how those subset components can be used to model responsibility charts.

In the original REA paper (1982, p. 562-563), McCarthy spoke of agent modeling in both a dynamic and static manner:

> As reflected in the classification schemes of a general ledger, the roles of participants in the economic affairs of an enterprise are accounted for in dual fashion. First, in a dynamic manner that involves parties both inside and outside of the company, specific *participation* in economic events is recorded. This application is reflected in the use of organizational unit codes for many expense and asset accounts and in the use of subsidiary ledgers for both receivables and payables. Second, in a more static manner that involves only inside parties, *responsibility* for the economic actions of subordinates is recorded. This application is reflected by incorporation of organizational responsibility charts into coding of accounts.

As a matter of easier explanation, the dynamic participation aspect mentioned above will be simplified in the rest of the monograph by using just the top level of the agent generalization hierarchy ("Economic Agent"), even in cases where being more specific might be possible. Usually, decomposition in the generalization plane is warranted by differential attributes, differential participation in associations, differential multiplicities, or differential methods. Again, for ease of explanation, we will generally not use these cases in our examples, although they are common in actual modeling use.

However, for the static case, we will give just one detailed example of an REA application at MOF Level-M1 in this appendix. Readers can infer from that case how those more detailed components would be used in larger-scale examples.

Figure 1-14 is our responsibility chart example, and readers should note these two features.

1. On the left is a sample organizational chart with a *company* having multiple *divisions*, which in turn have multiple *facilities*, which in turn again have multiple *departments*. All the agents in this structure have been stereotyped as Economic Units with superior-subordinate relationships, again reflecting the organizational hierarchies that are described as needed in the microeconomic literature (Kroszner and Putterman 2009) of the theory of the firm.
2. On the right is a partial decomposition of a firm's employees in the generalization plane. In this limited example, the subtypes of "Cashier" and "Buyer" have been divided out. For illustration sake, we can assume that only cashiers have the attribute "bondingStatus" and that only buyers have the attribute "buyerRating," so the decomposition here would be done for inapplicable attributes purposes (Smith J. M., 1978). Again, for example purposes, we will assume that employees are assigned to a company or a division or a facility. However, only buyers, can be assigned to a department, thus giving a second reason (differential participation in associations) for separating out buyers.

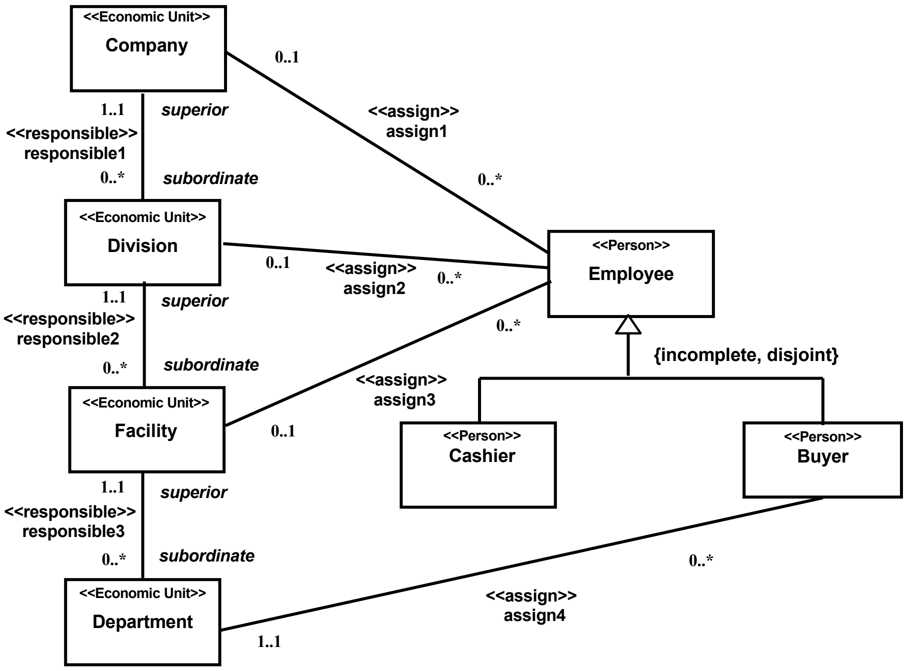


Figure 1-14 – An M1 example of REA Agent Subtypes with Organizational Economic Units

Again, this kind of organizational detail will not be warranted in further chapters, and we illustrate it here for completeness-of-example purposes. In subsequent chapters, employees will be stereotyped as Economic Agents and modeled with subtypes, such as cashier and buyer, without the detailed explanations given in this appendix.

Chapter 2: REA Expansion in the Granularity Plane

Specifying the Why and How of a Business Process

A business process is an occurrence in time that accepts resource inputs, uses those resources in directed activities, and produces resource outputs of value to a potential customer (Hammer & Champy, 1993, p. 53). REA was published some years before Hammer's seminal paper (1990) initiated the business process reengineering revolution, but the 1982 metamodel's essential infrastructure of matching decrement-increment economic events provides an ideal object pattern for a business process. The REA metamodel suggests that economic agents engage in business process activities, because these bundles of activities provide progress toward the definitive business goal of providing a portfolio of attributes appealing to a firm's ultimate customers. As mentioned in Chapter 1, modeling with resource-event-agent patterns provides answers to the **what, when, who, and why** questions of a business process definition. As we will see in this chapter, aggregating REA business process specifications into value chains and value systems (Porter 1985) provides an augmented answer to the **why** question, while disaggregating an REA process specification into workflows provides an answer to the **how** question.

Aggregating and disaggregating REA modeling components in the granularity plane can be compared to the zooming-in and zooming-out features of computerized mapping applications. If we use a *city* as a geographical starting point, then zooming-in would disaggregate the city successively into *neighborhoods*, *streets*, and *individual-addresses*, and provide detail on how the city was structured. Zooming-out would aggregate cities into *states* (or similar divisions like provinces/ territories) and *countries*, and provide detail on how the city was connected and related to its macro-environment. Somewhat analogously in REA modeling, we would start with a business process (i.e., the basic REA pattern shown in Chapter 1) and aggregate (zoom-out of) business processes to discover how they are aggregated into ***value chains*** (that is, the connections that take the firm's most basic inputs or resources and combine them into final outputs) and then are further aggregated into ***value networks*** (that is, the connections that create supply chains). Aggregating up gives us occurrences of a more coarse-grained or lumpier spatial granularity. On the other hand, zooming in on a business process would unearth its workflow structure as a disaggregation into collections of business events (that is, occurrences of a finer-grained spatial granularity). Business events are the steps that complete economic events.

Figure 2-1 portrays our starting point for a discussion of REA spatial granularity. It shows a shorthand notation developed by Geerts and McCarthy (1997b; 1999) to illustrate the REA metamodel (Figure 1-10) as a business process link in an overall value chain for an enterprise. This notation leaves inside economic agents as implied participants in the event components of a business process, and it leaves outside agents as components to be specified at the higher value network or supply chain level. The oval represents a business process where one or more input resources (such as labor, or material, or cash, or machines) are used or decremented purposefully in order to acquire or increment the output resource or resources (such as a finished good or a delivered product). For an REA application (MOF Level-M1), this business process notation is a shorthand method for showing the business plan for an enterprise.

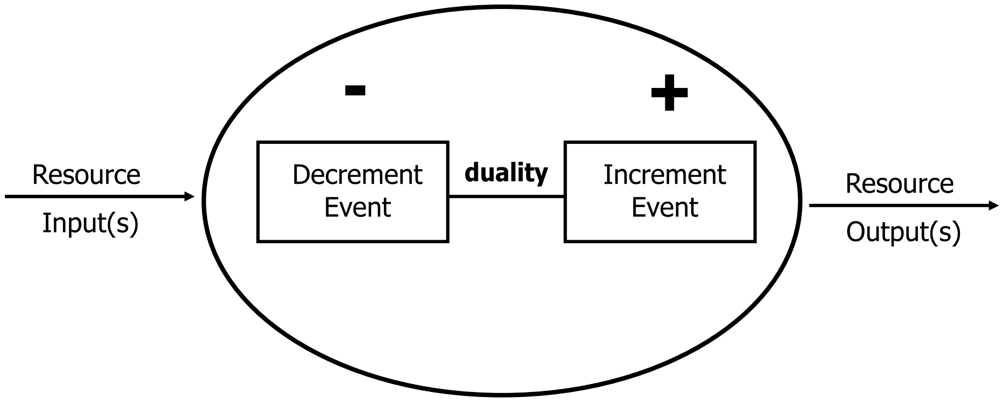


Figure 2-1 – REA Shorthand Business Process Notation

Aggregating and Disaggregating REA Business Processes

Value Chains [A00035]

Enterprise value chain is a concept popularized by Michael Porter (1985), but for purposes of discussing REA aggregation, a better description comes from Hergert and Morris (1989, p. 183):

> A fundamental notion in value chain analysis is that a product gains value (and costs) as it passes through the vertical stream of production within the firm (design, production, marketing, delivery, and service). When created value exceeds cost, a profit is generated. This notion of value creation is derived from the economics of demand. Products are viewed as bundles of attributes (Lancaster, 1975) which can be configured in multiple ways to appeal to segments of consumers having diverse demand functions.

Figure 2-2 shows nine example business processes aggregated into a prototypical value chain (based loosely on an actual ERP implementation for a company). Readers will appreciate that this M1 model of an REA application is an aggregation of business processes, achieved by using the outflow of one process as the inflow to another. This combined use of duality (within process) and stockFlow (between process) associations in REA modeling is best explained by Geerts and McCarthy (1997b, pp. 97-98):

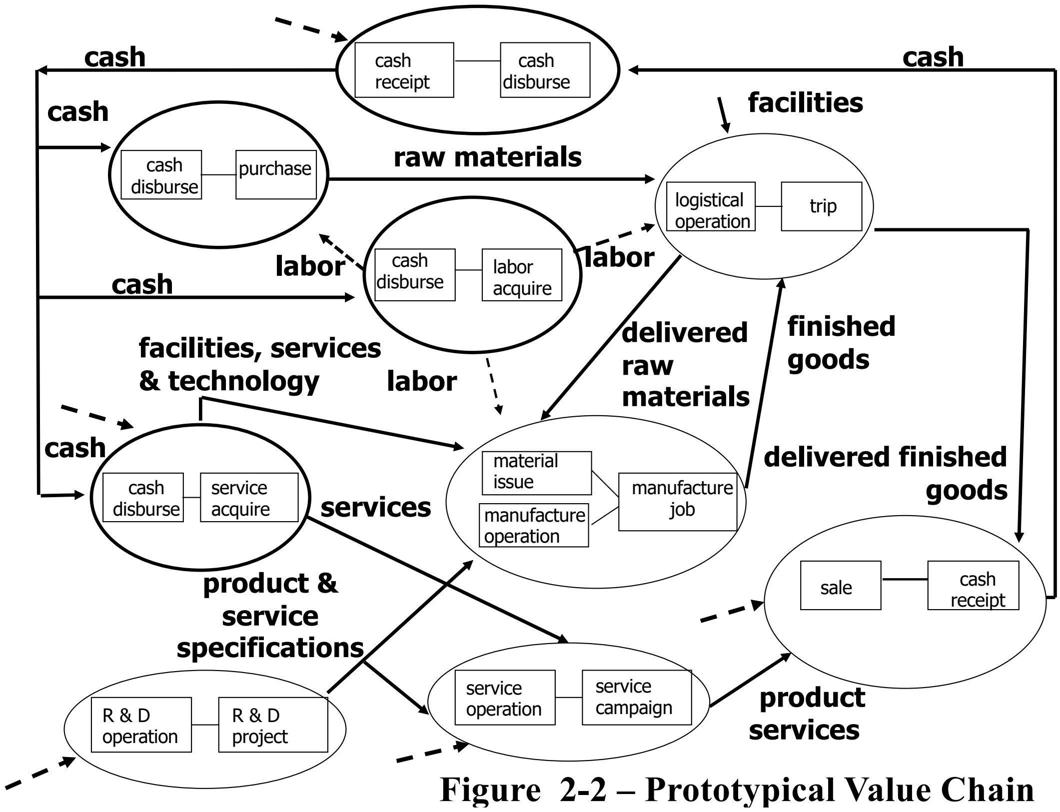


Figure 2-2 – Prototypical Value Chain

When all duality relationships are fully specified for an enterprise, the entrepreneurial rationale of its owner or manager (who are presumed to be *homo economicus*) is laid bare. No money is spent, or any other resource consumed unless an identifiably more valuable resource is acquired in return. Taken as a whole, duality relationships are the glue that binds a firm's separate economic events together into rational economic processes, while stock-flow relationships weave these processes together into an **enterprise value chain** … In its most general form, a value chain … is a purposeful set of economic exchanges where an initial outlay of cash is successively converted into some types of more valuable intermediate resource and then finally converted back to cash.

Figure 2-2 illustrates some conventions developed for use with REA value chains.

- The value chain initially concentrates on major resource flows at a very high level of aggregation. Figure 2-2 for example shows nine processes, and it fits on a single page to facilitate initial overall understanding. Each process could be disaggregated further. For example, the financing business process (shown at the top of the figure) could be disaggregated into debt-financing and equity-financing.
- Labor resources (output of a normal payroll process) are shown with dotted lines and disbursed to nearly all other processes, as employee labor is a key input for nearly all business activities. Space precludes labeling all labor flows.
- Representation of claims (imbalances in duality associations) is not usually exhibited at the highest value chain level. Claims like debt and equity (McCarthy 1982) usually warrant more detailed explication, a matter we save here for later explanation in Chapter 6.

In an actual enterprise the nine processes of Figure 2-2 would have to be decomposed further before detailed process representation and process engineering for a detailed ERP implementation could begin. The general structure of such a decomposition is illustrated in Figure 2-3, as adapted from Geerts and McCarthy (2001, p. 92).

At the top of Figure 2-3, an enterprise value chain is portrayed as a series of connected inflow-outflow processes. Each business process is adding value by converting resources into more valuable (to customers) resources. The new resource is then used as input by another business process. The business process decomposition layer, in the middle of Figure 2-3, illustrates further decomposition of one of the business processes in the enterprise value chain. The decomposition consists of two "leaf" (i.e., not needing further decomposition) nodes on either side of a middle process that is decomposed yet further. Let us suppose that this middle box (starred) is a manufacturing process which can be further decomposed into three sub-processes -- set-up, assembly, and inspection -- with the first and third of these also being leaf processes. Further, we may suppose that the sub-process assembly can be decomposed into three more processes: combining, welding, and painting. This gives us an example for the three-level decomposition shown.

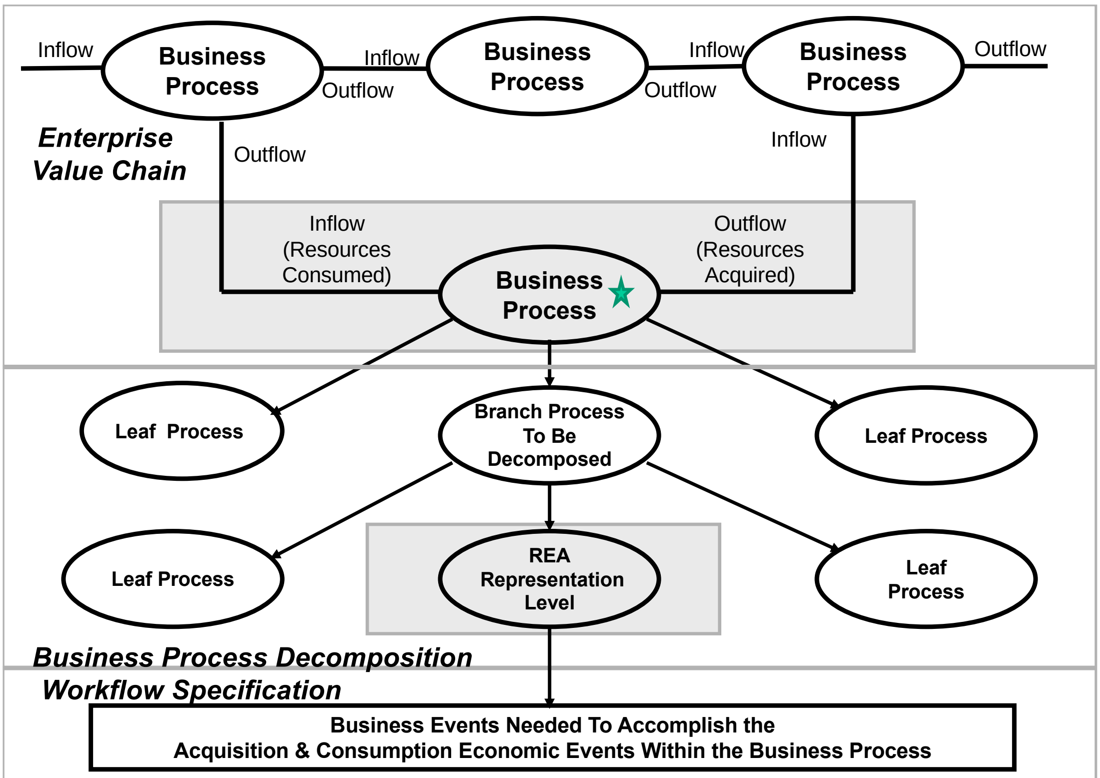


Figure 2-3-- Business Process Decomposition of Value Chain (Source: Adapted from Geerts and McCarthy 2001)

Such a multi-level decomposition would routinely leave an enterprise with 100-200 REA business processes to be implemented directly. Putting all of these business processes on one diagram would give a truer, more-detailed view of an enterprise value chain, but the less concrete examples like Figure 2-2 facilitate quicker overall understanding.

Workflow

Value chains *aggregate* business processes. At the very bottom of Figure 2-3, we see a description of further business process *disaggregation* as we dip into the finer-grained world of workflow with ***Business Events*** (initially titled as ***tasks*** by Geerts and McCarthy (1997b; 2001)). Business events are occurrences in time that management wishes to plan, control, and evaluate (David J. S., 1997; Denna, Cherrington, Andros, & Hollander, 1993; International Standards Organization, 2007). These occurrences normally use some labor and other resources (i.e., they qualify as resource decrements), but they usually cannot be paired logically and somewhat immediately with an increment event that produces an identifiable and representable resource. Business events progress a business process by moving the major economic events of that process closer to completion. Davenport (1993, pp. 5-7) gives an augmented definition of business process workflow:

> [Workflow is] a structured, measured set of activities designed to produce a specified output for a particular customer or market. It implies a strong emphasis on how work is done within an organization, in contrast to a product focus's emphasis on what. A process is thus a specific ordering of work activities across time and space, with a beginning and an end, and clearly defined inputs and outputs: a structure for action. ... Taking a process approach implies adopting the customer's point of view. Processes are the structure by which an organization does what is necessary to produce value for its customers.

Figure 2-4 illustrates a possible workflow for the Chapter 1 acquisition example. There, in Figure 1-8, we specified two economic events: a purchase and a cash disbursement. Figure 2-4 shows the workflow necessary to accomplish that purchase (receive, inspect, accept, and acknowledge the goods) and that disbursement (receive and pay invoice). These six business events are illustrated with BPMN 2.0 (Object Management Group, 2011) in Figure 2-4, and readers should understand that such workflows are highly idiosyncratic, depending upon the particular firm's technology and process practices (for example, vendor-managed vs. just-in-time vs. purchase-to-stock acquisitions). Each of these six business events could also have been decomposed into more detailed business events (for example, receive raw materials could be decomposed into count-items, add-to-inventory, send-information-to-inventory-control, etc.), or combined into larger business events (like handle-raw-materials). Again the key identifying consideration for business event specification is determining the level at which management wishes to plan, control, and evaluate its activities (Denna, et al. 1993). In Chapter 6, we suggest an approach -- state-machine mechanics -- for determining the efficient use of workflow resources, but at this point, we simply show their documentation as disaggregated views of the REA business process level shown in Chapter 1. Besides BPMN, we could have alternatively used older workflow notations like system flowcharts or data flow diagrams. Both of these are less compatible with modern ERP use, but they are quite popular in AIS textbooks and in audit notation of Sarbanes-Oxley reviews.[12]

[12] Another important reason for us choosing BPMN 2.0 notation is that it contains execution semantics which enable the definition of computer-readable workflow specifications. As we will explain later in the monograph, this enables us to meet our goal of integrated business process and workflow specifications that can be reasoned with as part of accounting applications (Geerts & McCarthy, 2000a).

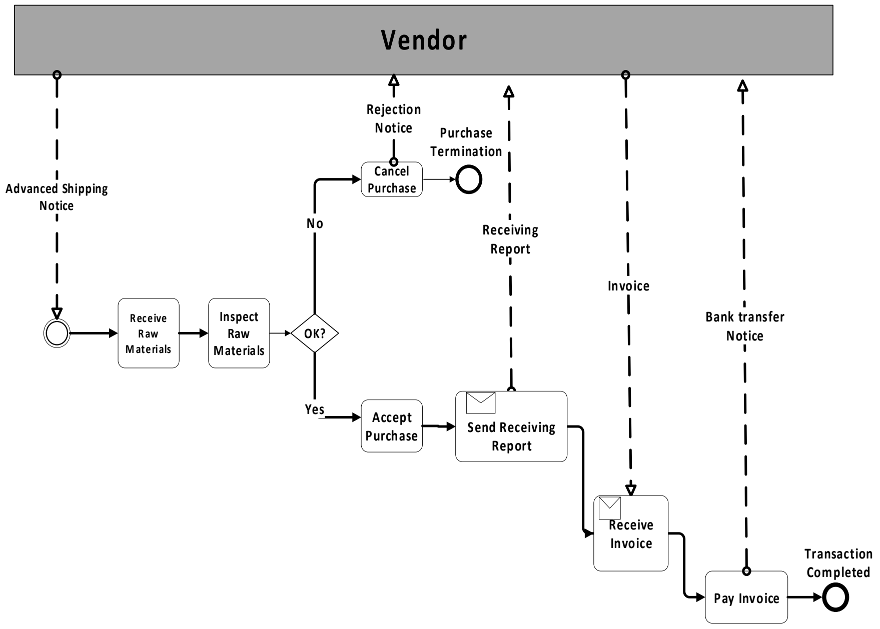


Figure 2-4 – BPMN Workflow for Raw Material Acquisition and Payment

Value Networks

Geerts and McCarthy (1997b; 1999; 2001) pioneered the notion of different granularity levels for REA models. They initially used three levels – value chain, business process, and workflow – but we expand that notation here to another level of aggregation – the value network or supply chain level, an additional combination suggested by Haugen and McCarthy (2001b). Value networks are akin to Porter's (1985) notion of value systems, an idea developed further by Dunn (2012), Haugen (2014), and Gordijn (2002). Our notation for value networks is illustrated in Figure 2-5 and explained below.

At the center of Figure 2-5 is illustrated a shortened value chain for an object enterprise with six business processes, surrounded by a solid box to indicate its scope. The object enterprise is surrounded by an example set of dotted-lined outside agents (a financier like a bank, two vendors like a raw materials supplier and a transportation provider, a set of employees, and two customers who buy the firm's products). Two of the example outside agents (E2 and E4) have 3-process value chains that use aggregated names suggested by the SCOR (Supply Chain Operations Reference) model (Zhou, Benton, Schilling, & Milligan, 2011).

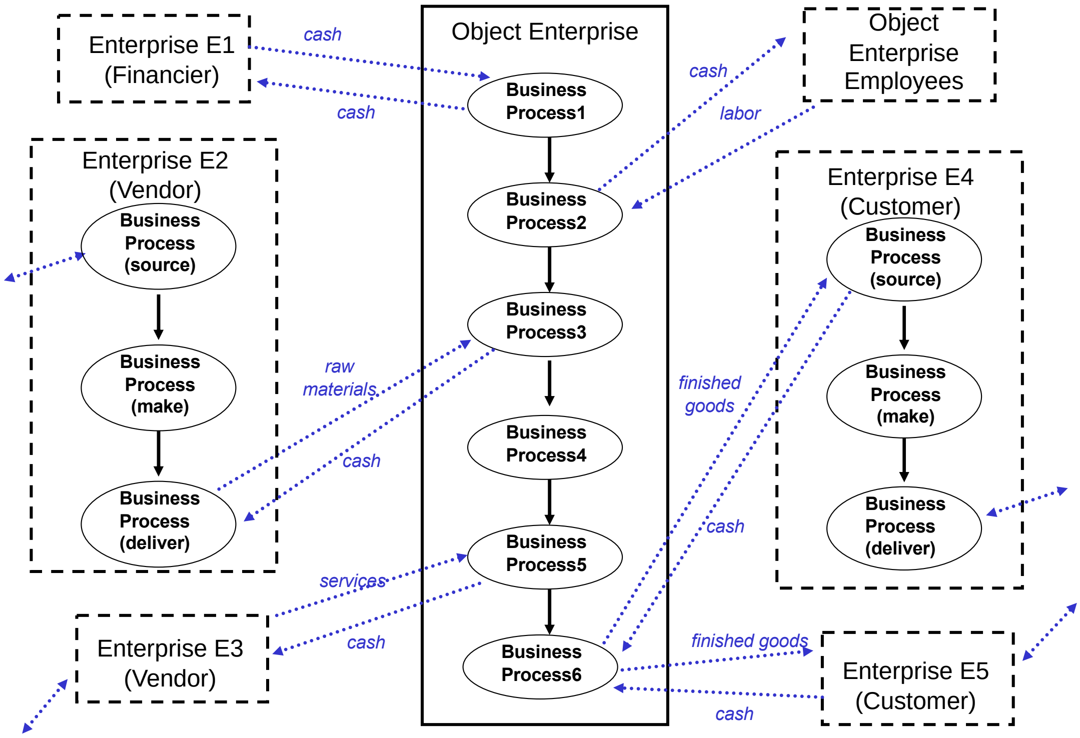


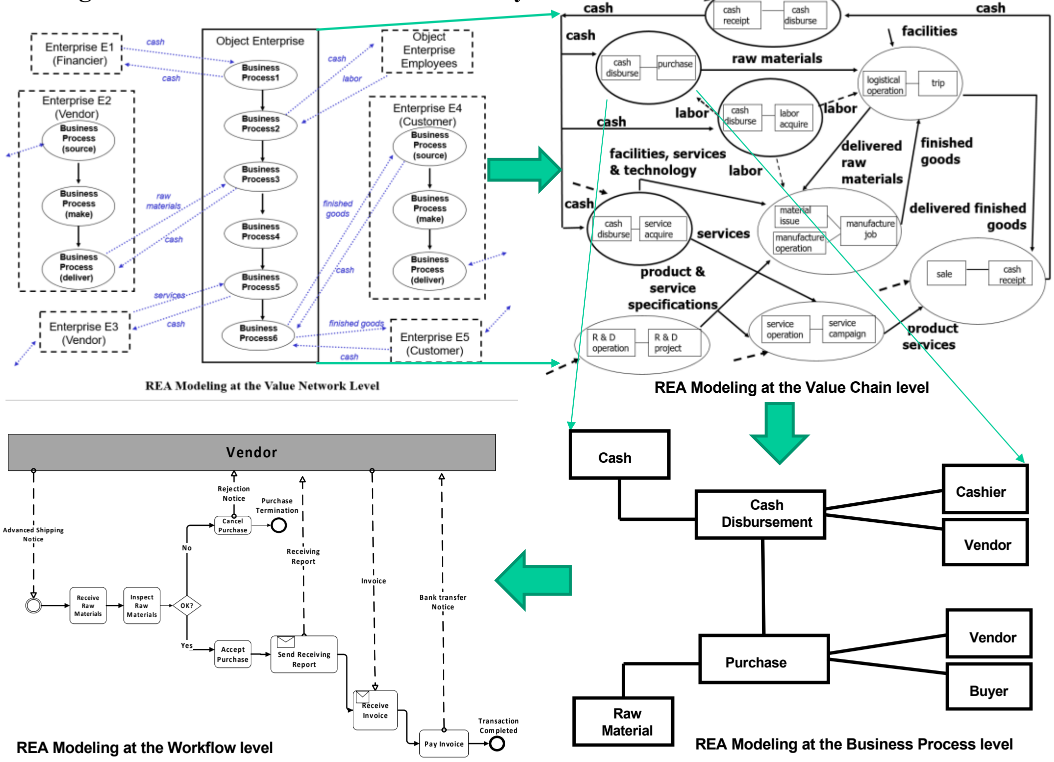
Figure 2-5 REA Modeling at the Value Network Level

As is true with all levels of REA granularity, the amount of detail to show for each value system component is subject to the judged level at which management plans, controls, and evaluates economic phenomena. For example, in analyzing a particular product component supply chain, the value network could include a more extensive set of outside agents with far less business process detail. At its most aggregate, the supply chain would consist of single boxes representing as many economic agents as was needed to explain where the factors of production for the object enterprise originate and where its final products conclude in use.

Portraying All Four Granularity Levels Together

In Figure 2-6, we illustrate our four levels of REA granularity as a combined version of Figures 2-5, 2-2, 1-4, and 2-4. Starting with the most aggregate level, we see that we can zoom in on the value network for an object enterprise (northwest corner) to uncover its value chain (northeast corner). Then, each one of those business process parts of the value chain (for example, its acquisition process that turns cash into raw materials) can be exploded into its resource-event-agent business process components (southeast corner), expressed here as a UML class diagram. And finally, we can zoom in on the economic events of a business process (for example, purchase and cash disbursement) to uncover its procedural workflow (southwest corner). For example, as specified in BPMN, a purchase is effected by receiving, inspecting, accepting, and acknowledging the purchased goods, while a cash disbursement is effected by receiving and paying invoices.

Figure 2-6 – 4-levels of REA Granularity



The Basic Elements of the REA Expansion in the Granularity Plan

In Chapter 1, we introduced REA terms that have been defined and/or used with slightly different connotations. Again, one of our main goals for this monograph is to formalize in a single location the primitives critical to understanding and to working with the REA ontology. With that purpose in mind, here are the new definitions for Chapter 2.

- ***Business Event*** – An occurrence in time below the granularity level of an Economic Event that consumes resources, but which cannot be paired logically and somewhat immediately with an acquisition or production event that materializes an identifiable and representable resource. Economic events usually comprise multiple business events, each of which can have either a single associated time (receive raw materials immediately) or a duration (inspect raw materials over the course of 10 minutes). The criteria for recognition of business events is the decomposition level to which the system modeler wishes to plan, control, or evaluate (Denna, et al. 1993) the economic phenomena under study. If a business event has duration, we automatically assume that it possess at least two sub-events: an instantaneous starting occurrence and an instantaneous ending occurrence (International Standards Organization 2007).
- ***Business Process*** – An aggregation of business events that accepts resource inputs, uses those resources in directed activities (business events), and produces resource outputs of value to a potential customer, or to a subsequent business process (Hammer and Champy 1993, International Standards Organization 2007). At a minimum, a business process contains two economic events: one an increment and one a decrement.
- ***Value Chain*** - An aggregation of business processes contained within a business enterprise or some similar organization. Resources passing through these business processes gain value and costs, and when the value chain creates more value than

costs, business organizations will generate profits (Hergert and Morris 1989, Porter 1985).

- ***Value Network*** – An interrelated network of businesses where organizations position themselves either to provide or to acquire resources via market exchanges. These organizations become connected by providing output resources that become input resources to a business process of another organization in the value network.
- ***Workflow*** – We use Davenport's (1993) perspective of a workflow that includes business events connected in a specific ordering with a beginning and end to the workflow. The specification of a workflow includes clearly defined inputs (and points of input) as well as outputs. Thus, a business process will include at least one workflow, usually with multiple business events.

Summary of REA Expansion in the Granularity Plane

Chapter 2 has explained how to aggregate REA business process models, as defined in Chapter 1, into value chains and value networks and how to disaggregate them by means of workflow specifications. While accounting remains its main focus, the extensions presented here widen REA's reach into areas such as business process reengineering, supply chain management, strategic analysis, and workflow management. All of this is done in an integrated fashion. REA's core unit of analysis is the business process, but this chapter showed how business processes can be linked by means of inputs and outputs to form value chains and value networks, thus conceptualizing how organizations create value. Further, workflow specifications formalize the specific activities assigned to the business unit responsible for the process.

Next, in Chapter 3, we discuss how REA's accountability infrastructure (*what is*) can be extended with policy specifications (*what could or should be*) and scheduling specifications (*what has been specified or reserved*), resulting in an expanded definition of business processes in the temporal plane.

Chapter 3: REA Expansion in the Temporal Plane

Commitments and Policy Specification

The REA model discussed in the two previous chapters defines the *accountability* infrastructure of an enterprise information system at different levels of granularity. **Accountability** is the traditional emphasis of accounting systems, tracking the past movements of resources between and within enterprises, and disclosing that transaction history in financial statements. However, as called for by McCarthy (1982) at the end of the original REA paper, many businesses require extensions to this original accountability infrastructure, of which *commitments* and *policies* are two important instances. The nature of these extensions to the REA accountability infrastructure is illustrated in Figure 3-1 and explained below.[13]

A ***commitment*** is 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 (Ijiri, 1975, pp. 130-137). Commitments therefore extend REA in the temporal plane, because they emphasize the scheduled future and the manner in which those scheduled events relate to economic activities that occur in the present.

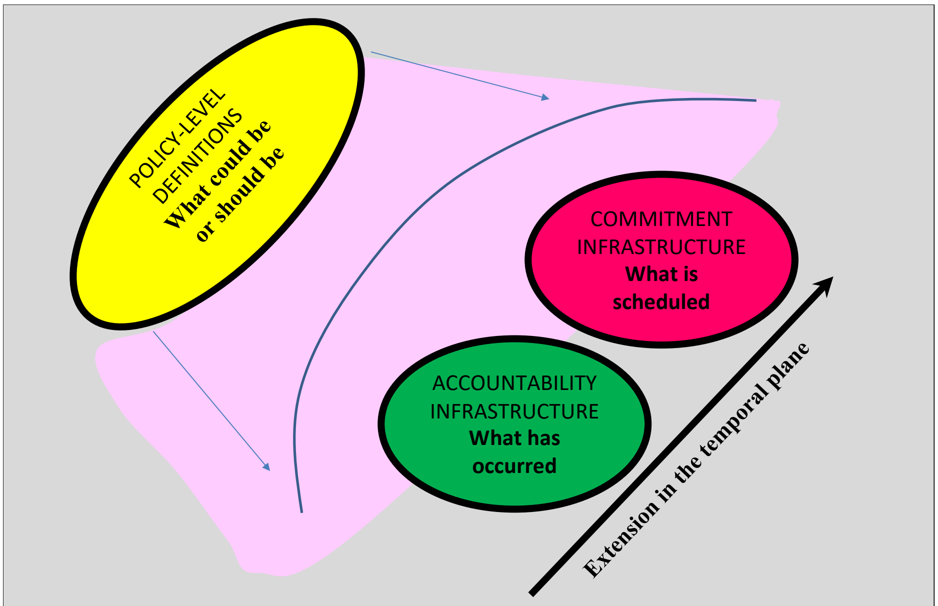


Figure 3-1 – Temporal Extension of the REA Accountability Infrastructure

A ***policy***, on the other hand, defines a guideline or constraint under which an enterprise operates: e.g., what could or should occur as sequences of activities unfold.

[13] Figure 3-1 foreshadows a categorization scheme (for REA metamodel components) that will emerge from this chapter. That scheme is specified and reflected in the final figure of the chapter (Figure 3-16). To control the complexity of that specification, we use a coloring scheme (adapted from REA teaching) to differentiate the temporal components of the REA metamodel. Thus, readers at this point should understand in a preliminary fashion these layers:

- the ***accountability*** layer – "what has occurred," as shown in green;
- the ***scheduling*** layer – "what has been specified or reserved," as shown in red; and
- the ***policy*** layer – "what could be or should be," as shown in yellow.

Guidelines define what to expect while constraints define permissible occurrences of the economic activities an organization engages in. As shown in Figure 3-1, policies apply to, and thus extend, both the accountability and the commitment infrastructures.

Policy definitions and commitment specifications heavily rely on advanced semantic structures which we label as type images and commitment images. These same structures form the conceptual basis for the advanced features of the Open-edi accounting and economic ontology – ISO 159944-4 (2007). Here we explain type images first.

Advanced Semantic Abstractions: Typing

The concept of typing goes back to Plato (Tarnas, 1991) where type images were described as ''archetypal forms'' by philosophers. For example, a horse is a real thing, while ''horseness'' is an abstract concept (Geerts & McCarthy, 2002; Sowa, 2000)that captures the archetypal essence of what it means to be a horse (like having four legs, a mane, and making a "neigh" sound). Much more recently, typing has been studied extensively by computer science researchers including Smith and Smith (1977a; 1977b), Sakai (1981), Fikes and Kehler (1985), Goldstein and Storey (1994), and Odell (1998). The typing semantic abstraction was first integrated as part of the REA model by Geerts and McCarthy (2006), upon which much of the following explanatory structure relies.

The primary use of typing is to capture concept descriptions that apply to a set of objects. Figure 3-2 displays a simple typing example at MOF levels M1 (class diagram) and M0 (informal animal images). AnimalType is a type image of Animal, because it extends the different attribute values of an individual like Traveller (a gray horse weighing 1700 pounds) with the typed attributes of "horseness" (a neigh sound). The dog Stevie and the lion Simba are similarly extended. Definitions (type specifications) apply to a set of instances (objects).

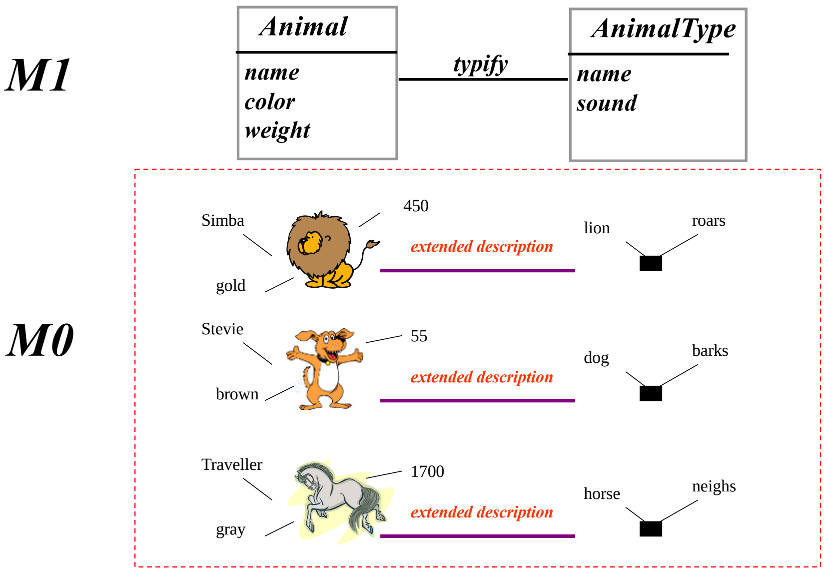


Figure 3-2 – A Typification Example

Figure 3-3 illustrates a typify association at the M2, M1, and M0 MOF levels for one of the REA primitives (Economic Resource): an aircraft as it is modeled for use by an airline company. An individual plane with the identifying code of "B93A14" has a fuel capacity of 6875 gallons plus a passenger capacity of 149 people (MOF level M0: rightmost column of

Figure 3-3). The latter two capacity characteristics, much like the neighing sound for Traveller, are attributable not to the individual instance of a plane, but to that plane's typed classification as a "Boeing737." In other words, all Boeing737s have standard fuel and passenger capacities as a result of their standardized manufacturing specification. In addition to an aircraft-type's standard fuel and passenger capacities, other attributes aircraft type might possess are "average years in service" – a computed total that ranges over all of the typed instances of that plane type – and "expected years in service." Very generally, type level attributes can be identified as being computed (e.g. "average years in service") or being essential or standard (e.g. "expected years in service" (Smith J. M., 1978; Smith & Smith, 1977a)). The classic Platonic archetype most clearly emphasizes the essential attributes, but modern computer structures accommodate both (Fikes & Kehler, 1985). For example, in object-oriented representation, both essential and computed characteristics are referred to as "static" attributes of a class (MOF Level 1). It is further indicated that all characteristics and rules for economic resources (MOF Level 2) apply to airplane types and to individual planes.

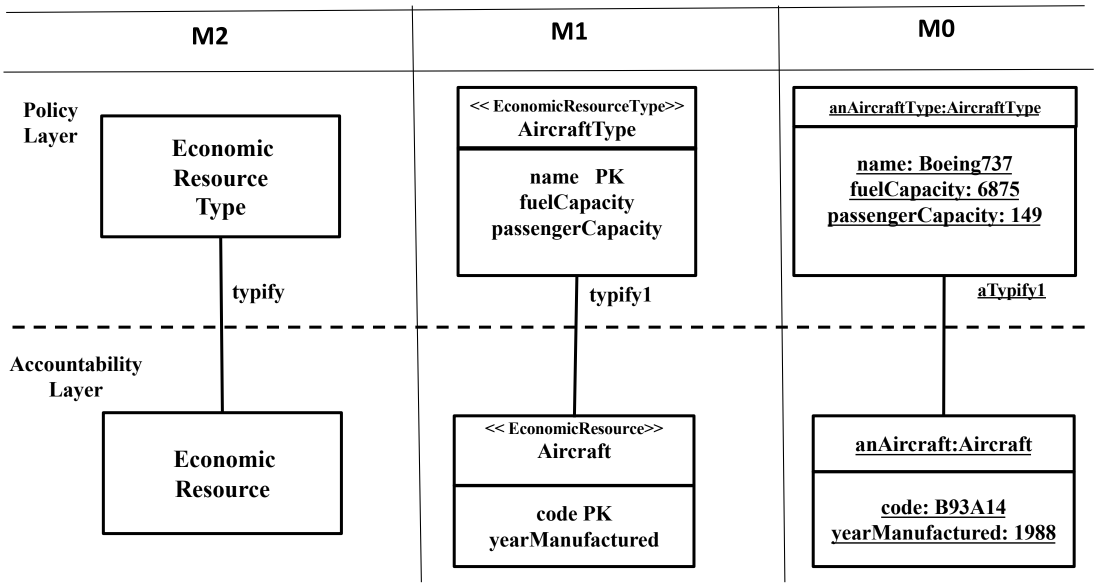


Figure 3-3 – Typification at M2-M1-M0 MOF Levels

Advanced Semantic Abstractions: Grouping

The concept of **grouping** works essentially the same as typing but lacks the Platonic pedigree of representing the archetypal essence of a set of instances. Grouping has been studied carefully by Goldstein and Storey (1994) and analyzed in further detail by Geerts and McCarthy (2006), upon which again much of the following explanatory structure relies.

An example of grouping is shown at the top of Figure 3-4: a set of houses that might be grouped into a neighborhood by a rental company. The neighborhood (a group) might have both essential attributes (like a rental surcharge for that neighborhood) and computed attributes (like the average yard size). However, in general, grouping tends to emphasize computed attributes over essential attributes, while the much more common mechanism of typing emphasizes the opposite. The differences between grouping and typing can be indistinguishable at the margin. For this presentation of the REA ontology, we simply consider grouping as a less common aggregation (Smith & Smith, 1977b) mechanism than typing, although it has clear implications for important accounting artifacts like budgeting (Geerts and McCarthy 2006).

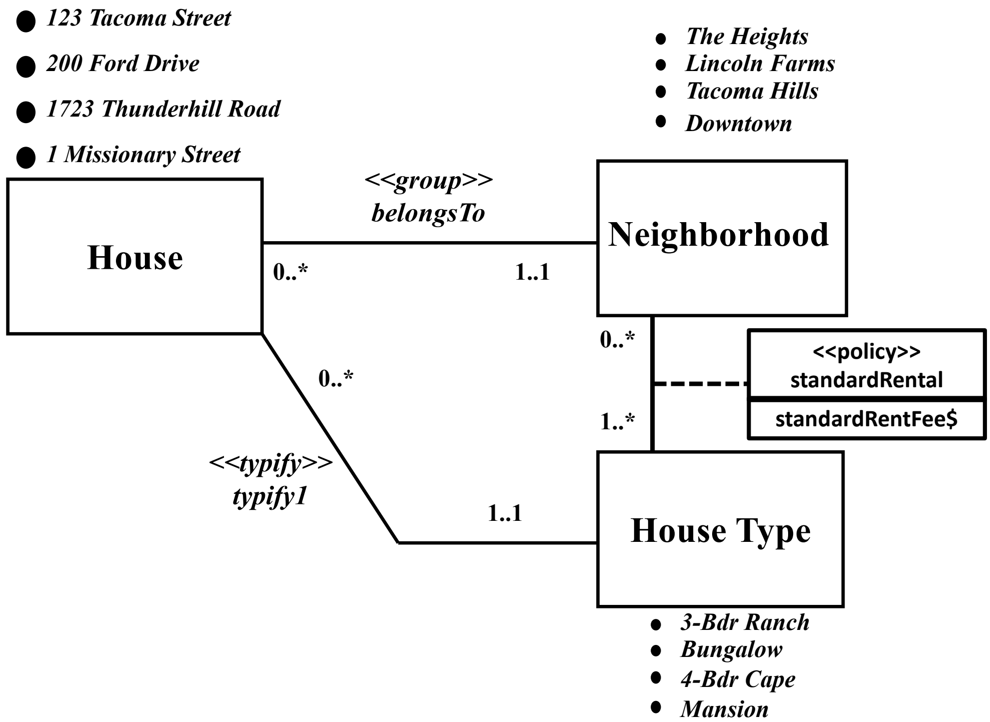


Figure 3-4 – REA Policy Layer with Typification and Grouping

Figure 3-4 further illustrates that typing and grouping can be used together. The top of Figure 3-4 illustrates the grouping of houses into neighborhoods. The bottom of Figure 3-4 abstracts the same houses to house types, a classification that could include essential characteristics such as minimum/maximum square-footage or a typical range of levels and bathrooms.

Advanced Semantic Abstractions: The Common Case of "Nail" Resources

Application of the typing and grouping semantic abstractions is not always as clear-cut as discussed above. An illustration is the common case in accounting of the definition of mass-produced products whose individual identification is technologically impossible at present, is economically infeasible, or is not perceived as meaningful by management. This is a very important exception that falls under the heading of something to be described in chapter 4: ***conceptual congruency***. However, to understand our later examples in this chapter where we discuss policy-level specifications and commitments, it is important to explain briefly this instance of congruency.

As instances of economic resources, Geerts and McCarthy (2006) divided tangible goods into two major classes: **car resources** and **nail resources**. Individual instances of *car resources* can be identified with a unique and universal characteristic, such as a vehicle identification number (VIN) for a car or an address for a particular house. Examples of car resources might include airplanes, expensive machines, and computers. *Nail resources* on the other hand, cannot be individually identified. Examples of nail resources would include mass-produced grocery items, chemicals or other liquid resources, and the raw materials and cash examples of Figure 1-8 in Chapter 1. Advanced technologies are changing the feasibility line between these two groups in favor of more nail resources being tracked individually. However, the distinction will certainly not disappear soon, so it merits explanation. Figure 3-5 illustrates the prototypes for which the two categories are named.

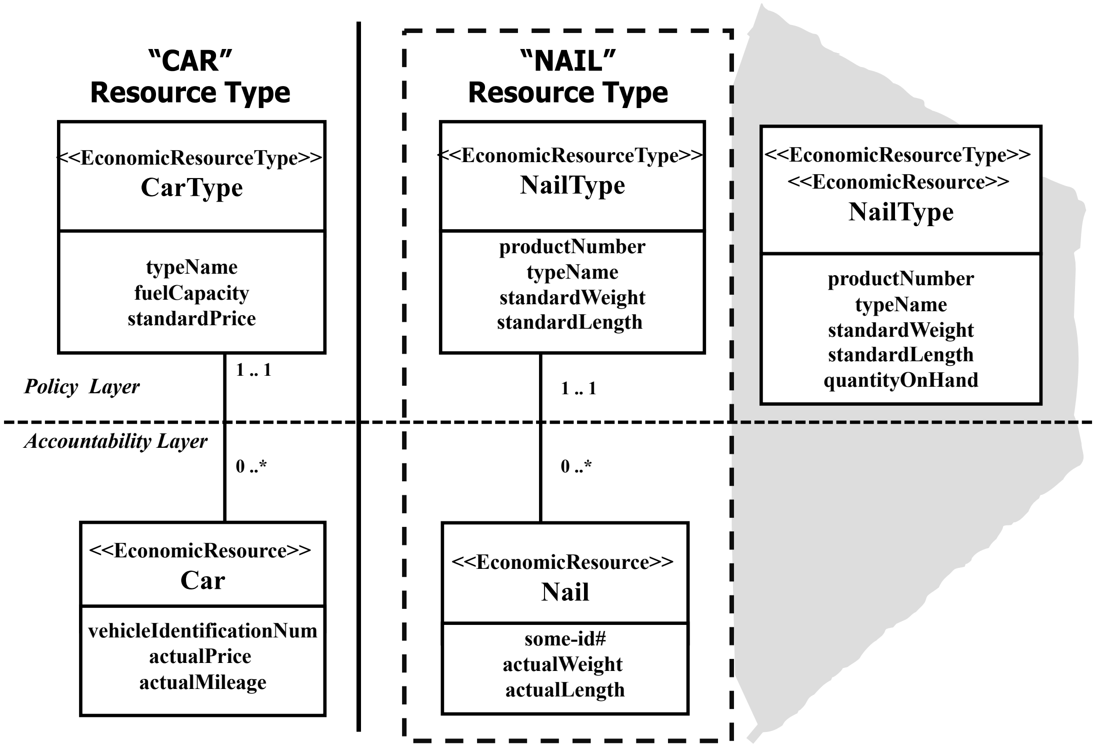


FIGURE 3-5 –"Nail" and "Car" Resources (Source: Adapted from Geerts and McCarthy (2006))

On the left are *cars* whose instances are readily distinguishable by vehicle identification numbers (VINs). Even two cars built to the exact same specification on the same day will have different VINs. At sale time, they could have different actual prices based on negotiation, and over time, they most certainly will possess different mileage values. However, their fuel capacities are determined by their type.

In the middle are *nails* whose class instances are almost always indistinguishable from each other, from both a physical and representational aspect. Nails of a certain type (3 inch galvanized) will most often possess the same product number at an enterprise like a hardware store, and it would be rare indeed for the actual weight and actual length of an individual nail to be recorded. It simply does not make representational sense or economic sense[14].

On the far right of Figure 3-5 is the representational compromise for nail resources which was first discussed by Geerts and McCarthy (2006) and which is further elaborated here. Because it is such a typical occurrence in commerce, this combination needs very specific handling. No information is recorded for the individual objects (nails) and the "Nail" object class is therefore not included (see transformation of the model in the middle to the

[14] One other example of a NailType resource would be gasoline in which there are no discernible individual units and therefore no prospect for tagging of instances.

model on the right). The hybrid class on the far right can be considered both an economic resource type and an economic resource, as indicated by the dual stereotypes shown. However, from an accounting perspective, information regarding how many specific nails are available, per type, is still needed. This can be done by introducing quantityOnHand as a computed attribute of the hybrid class. When the REA enterprise ontology is implemented in a completely automated sense, both of these roles will be specified: "EconomicResourceType" and "EconomicResource". However, for our discussion purposes in this monograph, we will usually just note the type image stereotype and explain within the discussion context how such a hybrid class can fill multiple roles.

Defining Policies by Connecting Type Images

A pioneer in the derivation of policy structures by combining type images was Clancey (1985) who developed his ideas under the AI (artificial intelligence) umbrella of "heuristic match," an idea that was later referenced in the taxation domain by McCarthy and Outslay (1989). Geerts and McCarthy (2006) extensively expanded these notions with both typing and grouping operations and applied them to the notion of accounting controls and policies. Connecting two types with a ***policy*** association is an extraordinary mechanism for specifying expected behavior in an enterprise. For example:

- An economic event type (like sales in certain amount ranges) can only be approved by certain economic agent types (like floor managers or store supervisors);
- An economic event type (like weekend or vacation car rentals) can only be offered to particular classes of agent types (like gold or silver status customers) and those rentals may also be restricted to particular types of economic resources (like minivans or convertibles); and
- A certain class of economic resources (like dangerous merchandise) can only be delivered with specific classes of vehicles (like specially constructed trucks) as driven by special classes of economic agents (certified drivers).

The middle right side of Figure 3-4 provides a more detailed example of a policy specification that employs both the typing and grouping semantic abstractions. The "policy" association on the right defines a standard rent policy by a housing enterprise that pairs specific categories of house types (type definition) with specific neighborhoods (group definition) to compute standard rentals. For example, a mansion in The Heights would cost much more than a mansion in Downtown, but it could be cheaper than a 3-bedroom ranch in Tacoma Hills.

Figure 3-6 illustrates, at the M2 level, how policies are specified in terms of type definitions and then linked with the accountability layer. Resources, events, and agents – the original accountability classes – are all typified and then their type images are connected by policy associations. There are many variations of this generalized structure as is illustrated by the following:

- Although used less frequently, *grouping* is also a powerful mechanism for the definition of policies, as was illustrated in Figure 3-4.
- It is sometimes the case that policy associations are best specified between the accountability layer and the policy layer, instead of just among or between elements of the policy layer. A good example of this might be a standing quote from a specific vendor for a certain type of raw material.

- Policies can also be defined in terms of just accountability level classes, and they can also be specified with single attributes at either the policy or accountability levels.

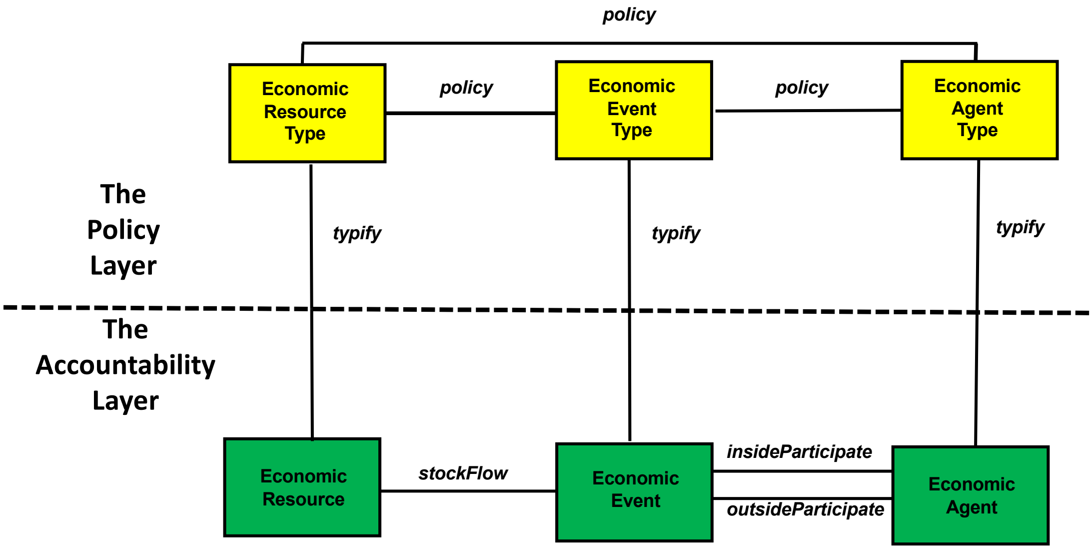


Figure 3-6 – REA Accountability/Policy Layer Interaction (M2)

Omission of such variations from the meta-model in Figure 3-6 is for reasons of parsimony. Or in the spirit of Occam's razor: we do not unnecessarily multiply entities in the construction of an explanatory theory of accounting data representation. Readers interested in a more extensive analysis of the variations explained above may consult Geerts and McCarthy (2006).

Figure 3-7 shows another more advanced M1 instantiation of the REA grammar (M2) for policy specifications defined in Figure 3-6. Readers should note these policy conclusions (although there are others that can be discussed):

- A flight type (DL 1729 which leaves Detroit daily at 10:00am, for Boston) has a standard kind (type) of aircraft assigned. Additionally, the aircraft commander to be scheduled for that flight type must have both a minimum pilot grade (a policy-policy association) and a qualifying number of hours flown in that kind of aircraft (a policy defined by an attribute).
- The flight type is identified by its flight number and it has a scheduled departure time and a scheduled duration. The actual flight is identified by the flight number plus the date, and it has an actual departure time and an actual duration for both accountability and variance analysis purposes.

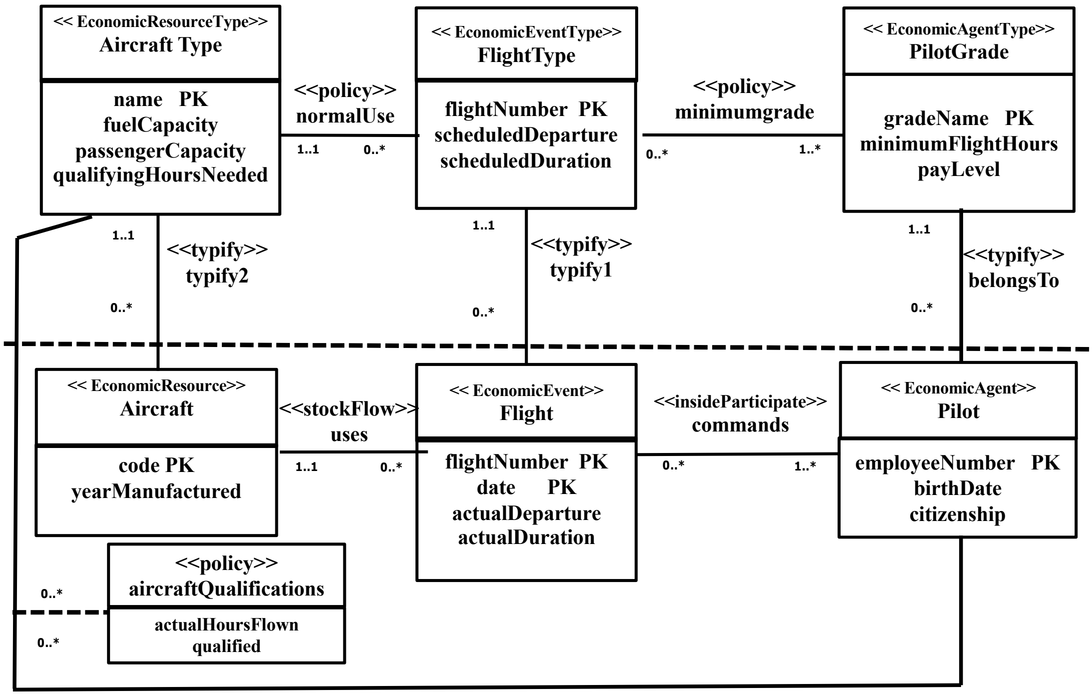


Figure 3-7 – REA Accountability/Policy Layer Interaction (M1)

Policy-layer constraints and standards apply most particularly to the accounting domain, because they can be used extensively in the specification of internal controls for auditing and standards for managerial accounting. Our treatment of policies completes the discussion of typing and grouping. We now move on to an explanation of how commitments and contracts expand REA in the temporal plane and how their specifications heavily rely on typing and grouping specifications.

Commitments and Contracts

The original REA work on commitments was done by Geerts and McCarthy (2000b; 2002), but many of the structures exhibited here have been influenced by the ISO project – ISO 15944-4 – undertaken to develop an accounting and economic ontology (ISO 2007). In turn, much of that work originated in efforts from the original ebXML (electronic business extended markup language) consortium (Organization for the Advancement of Structured Information Standards, 2006).

As discussed earlier in the chapter***, Economic Commitments*** are promises to execute economic events, and REA terms the named association between these occurrents as ***fulfill*** (i.e., events fulfill commitments). Commitments usually occur in pairs with a party promising to give up certain resources with the expectation that the other party promises that person something of value in return. Hence the relationship between economic commitments – ***reciprocal*** -- mirrors the relationship between economic events -- **duality**. The symmetry of these structures is apparent in Figure 3-8 which portrays at the M2 level the essential infrastructure of the REA scheduling layer.

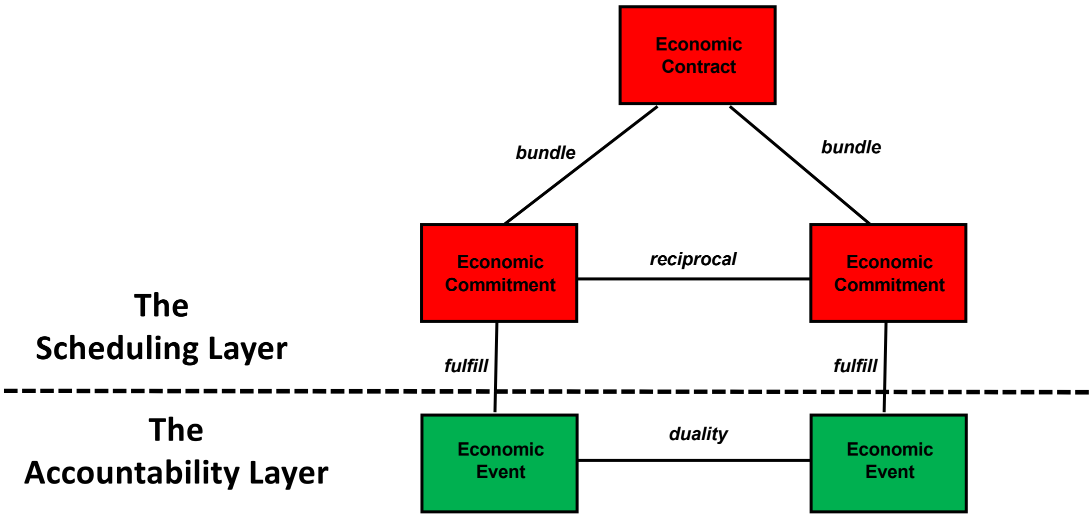


Figure 3-8 – REA Accountability/Scheduling Layer Interaction with Symmetrical Event and Commitment Images -- M2

Simple commitments are sometimes adequate for uncomplicated exchanges: a promise is given and a promise is taken. However, it is more normal that sets of matching commitments are bundled in an ***Economic Contract***. Even simple contracts often contain contingency patterns (Fowler, 1997) that detail what is scheduled to occur when a normal business exchange goes awry. For example, in a simple sale, what happens when delivered goods are damaged or late, or what happens when a scheduled payment fails to materialize. Each of these contingencies would represent a separate path through the scheduled business process, and each path would require an additional set of matched commitments in the contract. In object-oriented design circles (again, one of the foundation communities for the science of knowledge representation), the primary intended set of commitments is called the "Happy Path" (for example, the promised goods arrive in pristine condition on schedule and the reciprocating check clears immediately). The classes and associations illustrated in Figure 3-8 accommodate both the happy path and unhappy path,[15] mirroring the demonstrated need in traditional bookkeeping for accounts like sales-returns and sales-allowances.

Figure 3-9 illustrates an example of a common contract used in the automobile industry (where it is sometimes referred to as "PO1") for just-in-time coordination of raw material acquisition.[16] We use this case as an example of REA commitment structures throughout the rest of this chapter, unveiling in stages the different classes and associations for the scheduling layer. Readers should note that Figure 3-9 is an M1 example (its happy-path) for the M2 grammar defined in Figure 3-8 except that it shows additional accountability details with classes and associations for its economic resources. Both of the resources for "PO1" are nail resources (cash and raw material), so they are stereotyped as economic

[15] Unhappy path examples will be discussed in more detail in chapter 4.
[16] "PO1" is a M1 example from the automobile industry, which differs in some detailed ways from our cookie raw material purchase example in Chapter 1.

resource types.

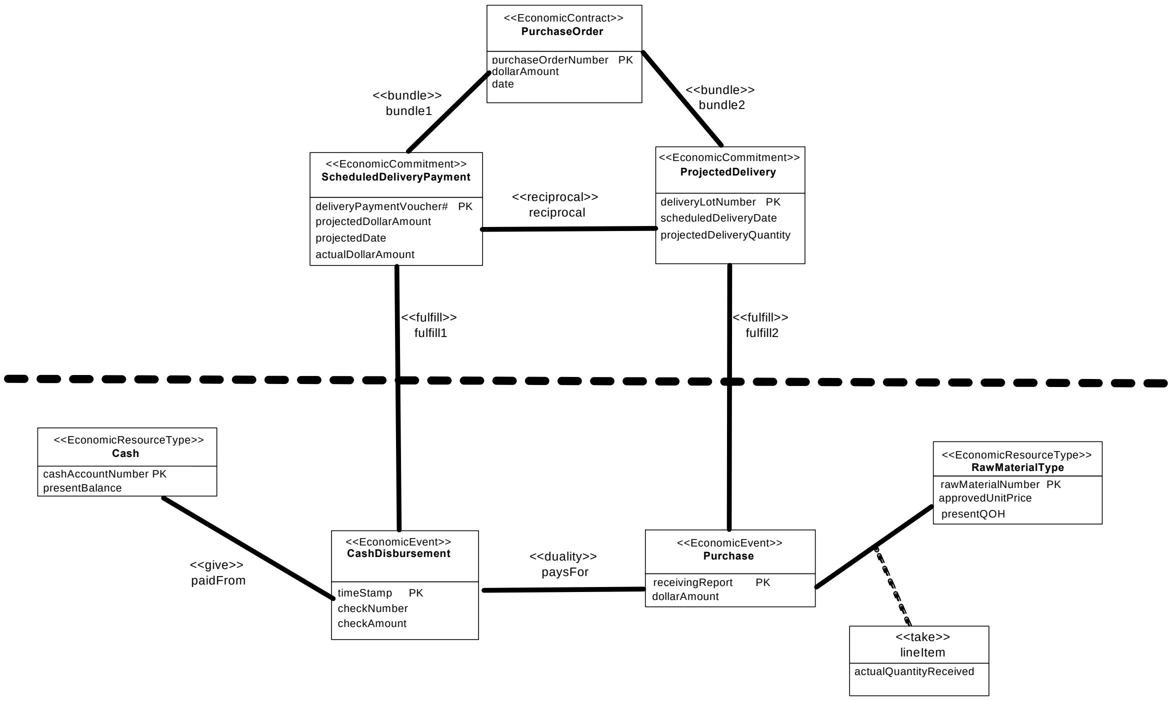


Figure 3-9 – REA Accountability/Scheduling Layers with Resources (M1)

Figure 3-10 is also an M1 example for the scheduling infrastructure of Figure 3-8, except in this case, we illustrate economic agents instead of economic resources. Our goal here is to show the role of agents as participants in economic contracts – a purpose for which the REA ontology introduces four new types of metamodel associations called ***insidePromise, outsidePromise, insideNegotiate, and outsideNegotiate***.

- Just as there are inside and outside agents for economic events, so this pairing exists for economic commitments. The ***promise*** agents for a commitment are the two independent parties (***inside*** and ***outside*** the enterprise) who execute the commitment.
- At a higher level of responsibility, the ***insideNegotiate*** and ***outsideNegotiate*** associations connect the contract to the inside and outside parties who negotiated that contract and the contingent bundle of reciprocal commitments associated with it.

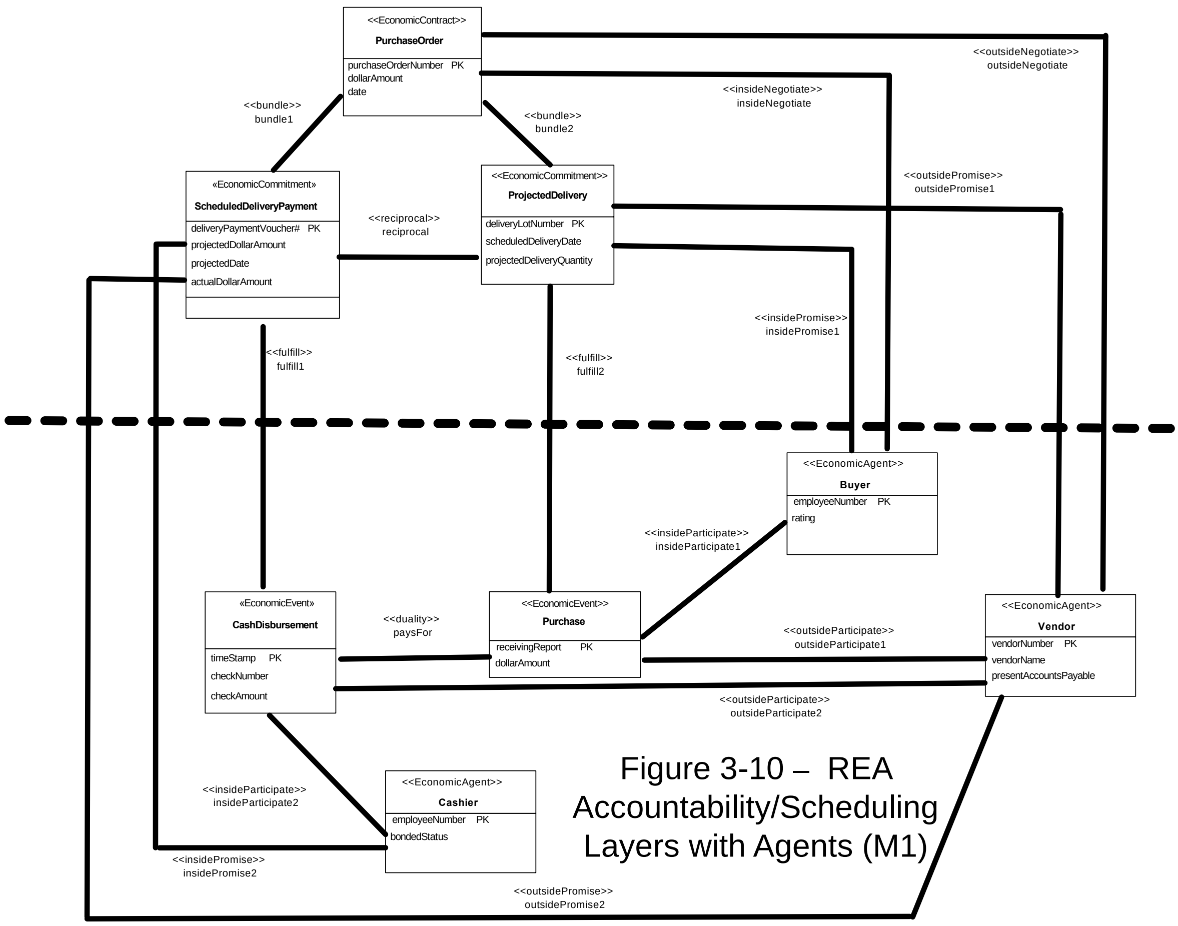


Figure 3-10 – REA Accountability/Scheduling Layers with Agents (M1)

In terms that are used for COBIT (IT Governance Institute, 2007; ISACA, 2019), the REA ***promise*** connection to commitments is similar to COBIT "accountable" (i.e., the person who provides direction and authorizes an activity) while the REA ***participate*** association with economic events is similar to COBIT "responsible" (i.e., the person who gets the task done).

Linking Commitments across Business Processes

Reciprocal associations bind commitments *within* a business process. Less commonly, the REA ontology also includes ***trigger*** associations to link commitments *between* business processes. This grammar extension is shown in Figure 3-11 at the M2 level. Trigger associations are especially useful in dependent demand supply chains where just-in-time (JIT) inventory policies prevail. Our "PO1" example is such a case as illustrated in Figure 3-12 at the M1 level. In dependent demand, resource commitments actually flow in the opposite direction from an enterprise's value chain – e.g., sale orders from customers trigger the need for production orders within manufacturing, which in turn trigger the need for purchase orders to vendors. If extended to the right, Figure 3-12 could also show the link between the conversion process and the revenue process.

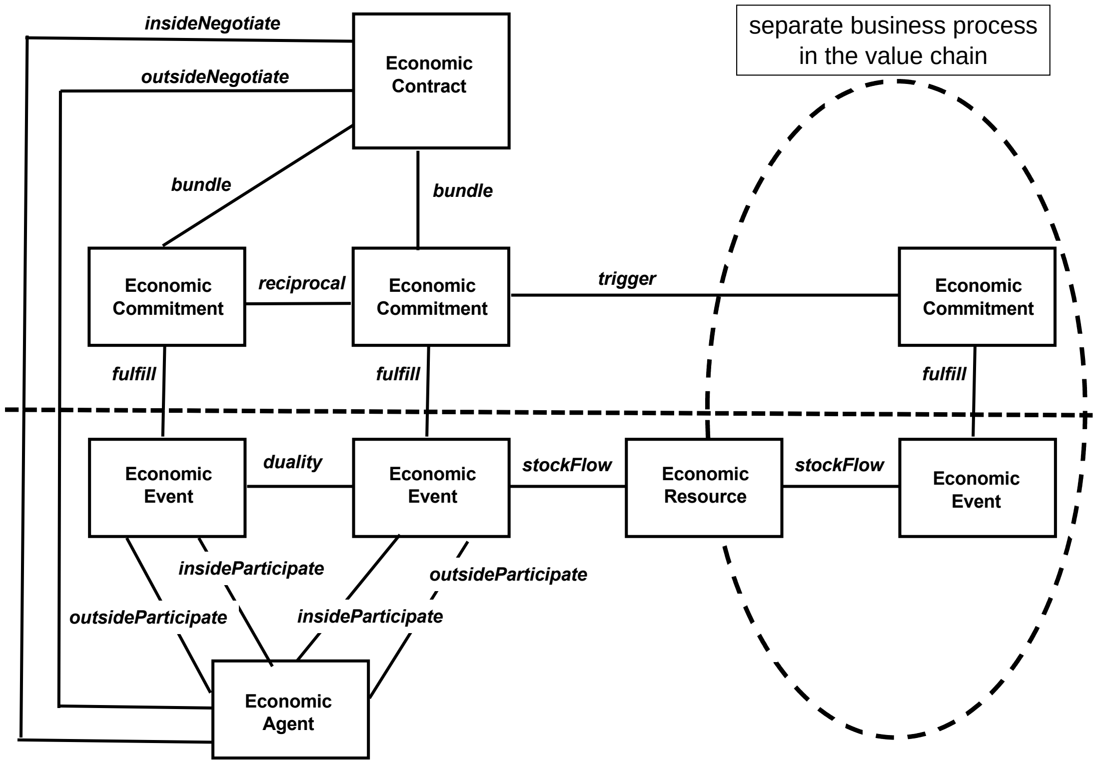


Figure 3-11 – REA Accountability/Scheduling Interaction Across Business Processes (M2)

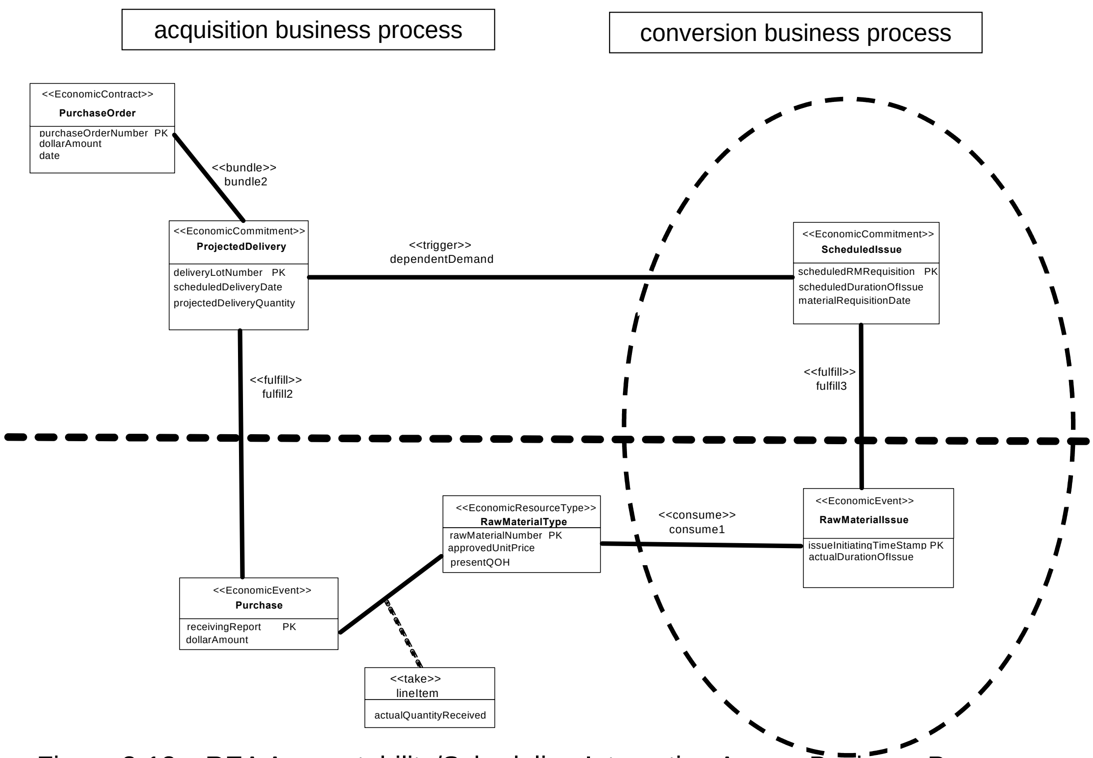


Figure 3-12 – REA Accountability/Scheduling Interaction Across Business Processes (M1)

Connecting the Scheduling Layer and the Policy Layer

When a promise is made as part of a contract, the two parties need to agree on the nature or type of the resources to be received by each. Less commonly, they might need to agree on the type of the economic event (for example, a retail sale vs. a wholesale sale) or the type of the economic agent (for example, a dentist vs. a hygienist for a dental appointment). The REA grammar (M2) for such specifications is shown in Figure 3-13. ***Specify*** is used to categorize the associations between a commitment and the types of resources, events, and agents involved in that commitment. As stated above, the most common commitment specification in REA is for economic resource types. Readers should also note that the presence of strongly-typed policies in a particular enterprise might reduce the need for complete REA specification as shown in Figure 3-13. For example, in a medical facility, if a certain type of appointment required (by policy) a certain type of medical specialist and the use of a certain type of medical equipment, then the appointment commitment would only need to specify the event type.

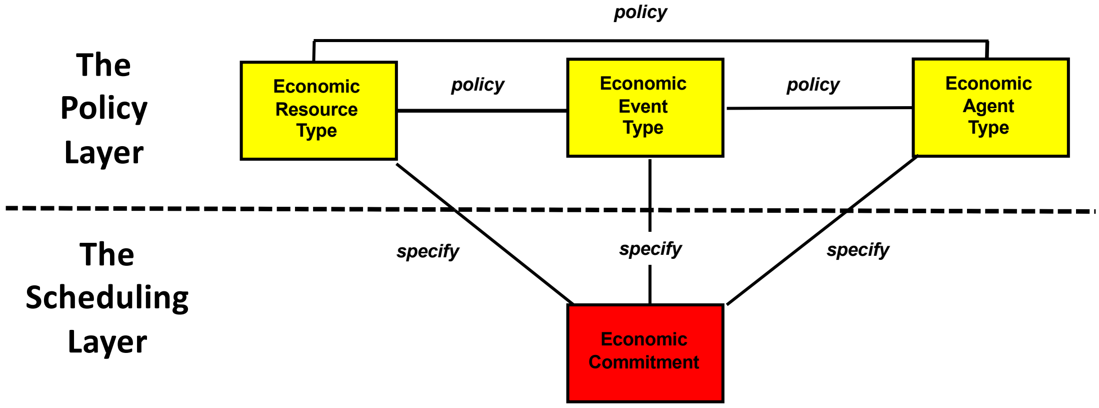


Figure 3-13 – REA Scheduling/Policy Specification of Type Images for Resources, Events, and Agents (M2)

Figure 3-14 illustrates an M1 specification for part of our "PO1" example; which applies the ***specify*** association to the two needed resource types. It is especially interesting to note how "PO1" commitments and policies work together. As we saw in Figure 3-12, a scheduled issue in manufacturing "triggers" a projected delivery (a commitment). In Figure 3-14, we see the continuation of that scenario where the projected delivery "specifies" a certain raw material type. This specification is actually extended by the recursive "policy" association on the raw material type class. The full contract language of "PO1" stipulates that if a specified raw material is not available, then there is a ranked list of accepted substitute raw materials that can be used instead (a policy).

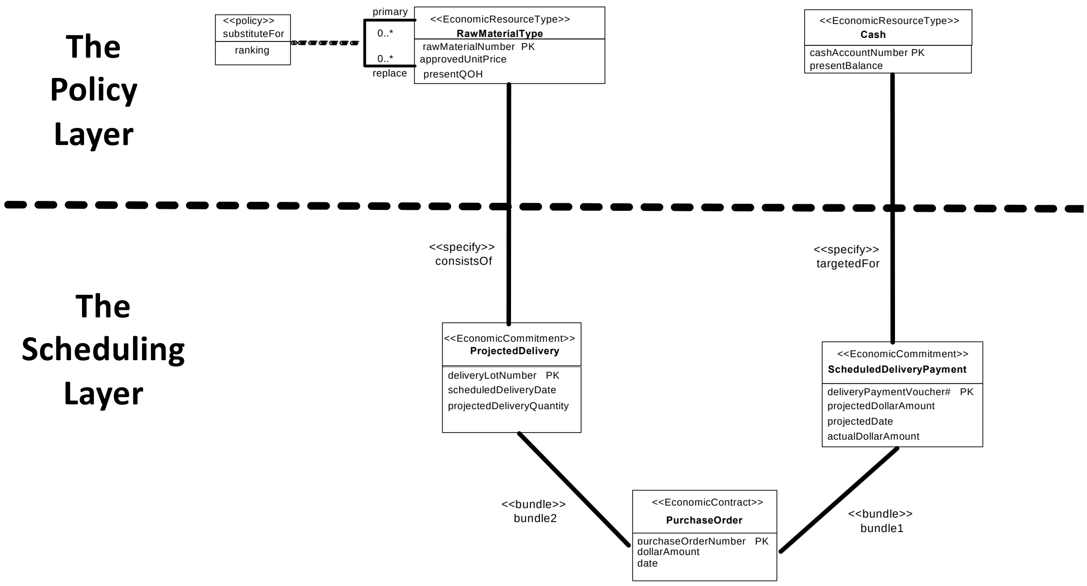


Figure 3-14 – REA Scheduling/Policy Specification of Type Images (M1)

Abstract specification of resources, events, and agents is undoubtedly the norm for contracts. There are however circumstances when a commitment reserves a specific resource or person instead of an abstract type. In the case of a real estate resources for example, it is normal to reserve the particular address rather than a house type or an apartment type. In the case of an agent, it might be the case that a factory supervisor reserves a particular carpenter for a future manufacturing run, rather than a "qualified carpenter slot". Use of the ***reserve*** association is illustrated in Figure 3-15.

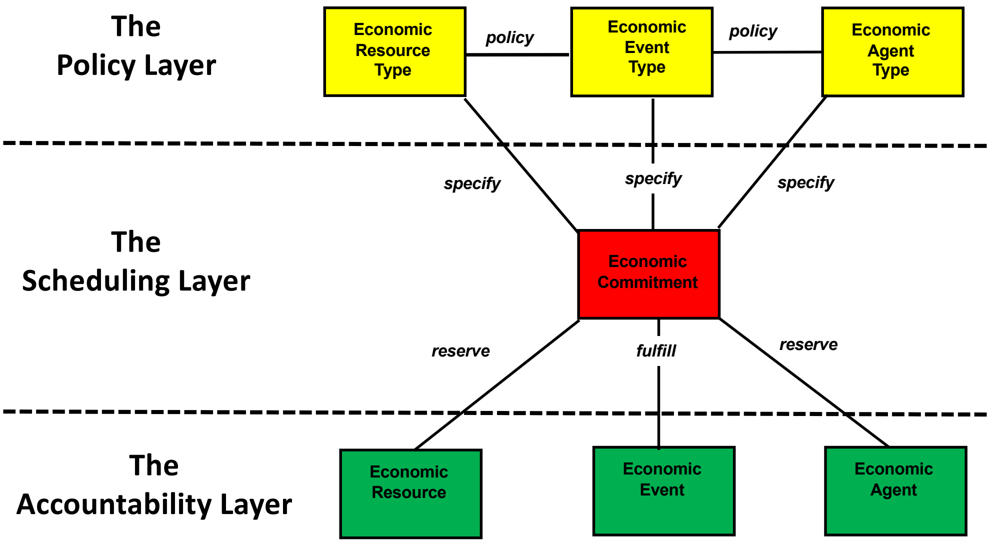
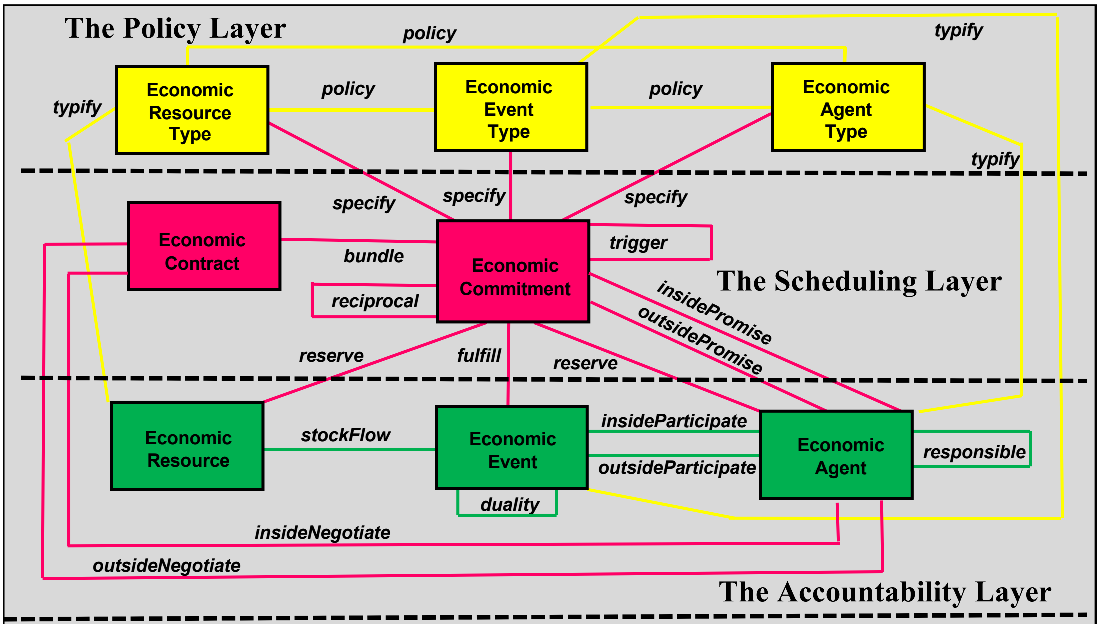


Figure 3-15 – REA Accountability/Scheduling/Policy Specification and Reservation (M2)

The REA Metamodel with All Layers Integrated

If we incorporate all the additional classes and associations discussed in this chapter under the heading of expansion in the temporal plane for REA, the accountability infrastructure of Figure 1-5 expands to the specification illustrated in Figure 3-16.[17] We finish this chapter by reiterating specific definitions for the new classes and associations introduced in this expansion.



The POLICY layer – "What could be or should be": REA Types; typify, policy

The SCHEDULING layer – "What has been specified or reserved": Economic Commitments, Economic Contracts; fulfill, specify, reserve, reciprocal, trigger, promise (inside, outside), negotiate (inside, outside)

The ACCOUNTABILITY layer – "What has occurred": Economic Resources, Events, and Agents; duality, stockFlow, participate (inside, outside), responsible

Figure 3-16 – The REA Metamodel (M2) for All Temporal Layers

Classes:

- ***Economic Resource Type*** - the abstract and extended specification of an Economic Resource where its essential, grouped, or standardized properties can be designated without attachment to a specific resource. This definition is adapted with slight modification from ISO 15944-4 (2007, p. 5).
- ***Economic Event Type*** - the abstract and extended specification of an Economic Event where its essential, grouped, or standardized properties can be designated without attachment to a specific actual occurrence in time. Again, this definition is adapted with slight modification from ISO 15944-4 (2007, p. 4).
- ***Economic Agent Type*** - the abstract and extended specification of an Economic Agent where its essential, grouped, or standardized properties can be designated without attachment to a specific person or unit. Again, this definition is adapted with slight modification from ISO 15944-4 (2007, p. 5).

[17] For simplicity purposes, roles have been omitted from Figure 3-16.

- ***Economic Commitment*** - 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. This definition is adapted from Ijiri's (1975) discussion of commitment accounting (1975, p. 130-37).
- ***Economic Contract*** - the agreement between two Economic Agents to a bundling of reciprocated Economic Commitments, each of which details the specific or abstract nature of the resources to be exchanged or converted. This definition is adapted with slight modification from ISO 15944-4 (2007, p. 4).

Associations:

- ***typify*** - the association between a concrete entity and the abstract specification of its essential and computed properties. Typifying a concrete object, person, or occurrence attempts to specify properties that capture its archetypal essence. This definition derives from Platonic philosophy.
- ***group*** - the association between a concrete entity and another concrete class that defines its aggregated and computed properties. Typing and grouping are abstraction mechanisms that at the margin are very close to each other. In the REA ontology, we consider grouping as that set of aggregation associations that lacks the archetypal essence validation.
- ***policy*** - an association that usually occurs between two typed or grouped objects where its intent is to capture the meaning of enterprise rules or standards. However, policy associations often also occur between abstract and concrete objects where again the intent is to capture the meaning of enterprise rules or standards. This definition partially derives from the heuristic match ideas of Clancey (1985).
- ***specify*** - the association between an economic commitment and the abstract properties of a typed resource, event, or agent.
- ***reserve*** - the association between an economic commitment and the concrete properties of an agent or a resource.
- ***reciprocal*** - the association between economic commitments where the promise of one economic agent to execute an economic event is reciprocated by another economic agent promising a requited economic event in the opposite direction. This definition is adapted with slight modification from ISO 15944-4 (2007, 9).
- ***trigger*** - the association between two economic commitments across different business processes where the occurrence of the first commitment leads to the occurrence of the second.
- ***bundle*** - the association between economic commitments and the economic contract that bundles those promises. This definition is adapted with slight modification from ISO 15944-4 (2007, p. 4).
- ***insidePromise*** and ***outsidePromise*** -- the associations that bind inside and outside Economic Agents to the terms of an Economic Commitment. Commitments are promises, and these two designated agents are accountable for executing those promises.
- ***fulfill*** - the association that connects an Economic Event to the Economic Commitment that promised its occurrence.
  insideNegotiate and outsideNegotiate - the associations that connect an Economic Contract with the Economic Agents who have defined the specific terms of the bundle of Economic Commitments in the contract.

Chapter 4: REA Business Process Extensions

Systematic Contraction and Expansion of the REA Metamodel

The idea that the REA metamodel is actually a "pattern" for business process analysis was explained by McCarthy (2003a, p. 400)

> In semantic database design (and also in its closely related analysis cousin of object-oriented design), the hardest step is always the first: coming up with a good list of candidate entities (or objects or classes) on which to base the rest of the analysis. … To overcome this difficulty, the *analysis patterns* movement was born (Coad, 1995; Fowler, 1997; Hay, 1996) in the early 1990s. The REA accounting model preceded this work by a decade, but its basic framework of interlocking constellations of Economic Resources, Economic Events, and Economic Agents was actually a complex aggregation of some of those patterns that surfaced in the nineties.

Most basically, a pattern is a stereotypical constellation of classes that gives a designer an idea of what objects or categories to look for when analyzing a particular piece of reality.

For introductory explanation purposes, we show in Figure 4-1 a simple non-REA example: a pattern that illustrates the stereotypical constellation of entities that one might expect in a model of a college.[18] If analysts were tasked with building a data model of a particular college, they could use this pattern before they actually visited or started to study that individual school. Upon actual physical inspection of the campus, the analyst might choose to add entities like science laboratories or to delete preassigned pattern components like dormitories if warranted. In any case, the pattern gives a good starting point for building a database model. The *components* of an analysis pattern are its proposed list of classes as connected in a semantic network by named associations, so the components of our college pattern are classes like student, faculty, department, course, course-offering, and associations such as faculty to departments (assigned to) and students to course-offerings (enrollsIn).

[18] Obviously, this pattern or frame could be much more complicated and detailed. The required level of detail could depend on the depth of the analyst's understanding of the domain or on the level of detail required in any resulting system.

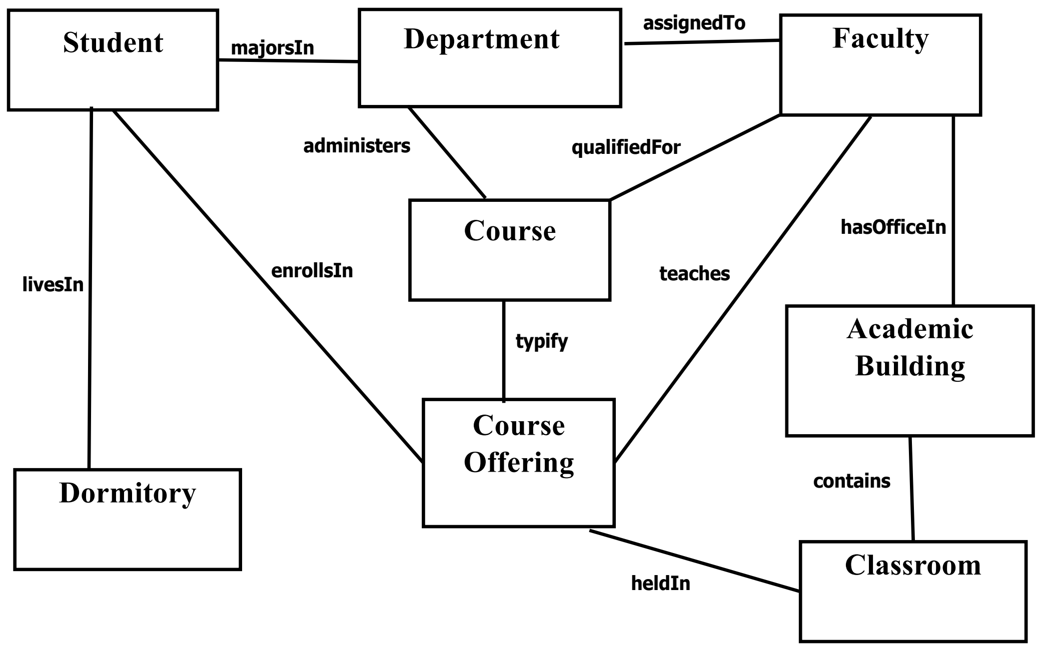


Figure 4-1 – Analysis Pattern for a College

As McCarthy expresses above, we consider the REA metamodel of Figure 3-16 as a pattern for a business process. The accountability layer of the metamodel constitutes the <u>normative</u> components of the pattern. This is actually the original 1982 constellation of classes, and by normative, we mean that a business process by definition <u>must</u> have the two symmetrical REA clusters (one an increment, the other a decrement) connected by a duality link. In contrast, the policy and scheduling layer components are <u>informative</u>, by which we mean that a business process for a specific company <u>may explicitly include</u> things like commitment and type images. For example, a cash purchase of items at a garage sale doesn't really need the involvement of commitments or types.

As noted by Brodie, Mylopoulos, and Schmidt (1984) and as emphasized later by McCarthy (1987), knowledge representation concepts and ideas in the two different computer science subfields of database design and artificial intelligence (AI) are sometimes very similar because of their common origin in their parent fields of philosophy, psychology, linguistics, and mathematics. The AI analog of an analysis pattern is the concept of a *frame* (Minsky, 1975) where components of the pattern are called *slots*. McCarthy (1987) used this strong correspondence to illustrate representation of the REA template as an economic event frame with slots for inside agent, outside agent, and resource. Applied to the REA metamodel of Figure 3-16, we could say (in artificial intelligence terms) that we have a frame for a business process with slots for components such as resources, events, agents, commitments, types, etc. We will from this point forward use the concepts of patterns and frames interchangeably.

In the rest of this chapter, we will discuss situations where components of the REA metamodel (or slots in the REA frame) are either expanded or contracted systematically in use, primarily because of representation constraints, but also because of business or industry practice. It is important to understand how these expansions and/or contractions work systematically for two reasons:

- so they can be accommodated or even reversed in the context of using REA for automated or intensional reasoning (Geerts & McCarthy, 2000a), one of the prime functions for a well-developed domain ontology; and
- so accounting system users and analysts can understand how full-REA systems must sometimes be contracted or expanded to meet system implementation constraints.

We attack the more complex issue of REA slot contraction first.

Contraction of Metamodel Components – Conceptual Congruency

"To be is to be the value of quantified variable". This is a famous quote of the philosopher Willard Van Orman Quine (1992) as reiterated and explained by John Sowa (2000, p. 52), a criterion to be used for admission of an ontological category into a representation scheme. In earlier work, Quine (1969, p. 23) had expressed this notion as "no entity without identity." In both philosophical cases, the import of the ideas is similar: It is difficult to represent and discuss a concept without providing a name for identifying the different individual instances of that concept.

For REA implementation, "no entity without identity" means that a candidate class for a particular model (at MOF Level-1) must have an identifying attribute that can stand for the instances of that class (at MOF Level-0). For a database, this means that all classes must have a primary key (PK) attribute (or combination of attributes) that is universal (every instance possesses that attribute) and unique (every instance's value for that attribute is different from all other instance values). Although it is sometimes possible that database and object-oriented (OO) implementations have system-generated primary keys, it is always the case that there is a domain relevant name for individual objects that can act as an identifier. For example, in an OO language like JAVA, instances or objects in the class CUSTOMER need differentiated names like firstCustomer, secondCustomer, etc. The requirement for an identifier is thus a representation constraint on an REA implementation. However, Rockwell and McCarthy (1999) devised a notion called ***conceptual congruency***[19] that mitigated the effect of this constraint on the process of developing an REA-compliant implementation. The idea of conceptual congruency was discussed in several places in the original REA paper, and it has already been mentioned in chapter 3, but we explain it more completely here with multiple examples.

In Figure 4-2, we illustrate generically how conceptual congruency works. Two classes and the association between them are simply combined into one class, because the set of instances postulated for each are deemed to be (eventually) congruent. The one class can then fill two slots in the REA frame with the list of attributes for those classes describing characteristics of both components. For example, a reservation for a hotel could have "reservation#" as an identifier, and the same class could actually represent both the commitment for a stay in the hotel and the actual stay itself once a customer arrives. The congruent class could have two attributes (among others) describing both slots, like "planned number of days for the stay" and "actual number of days for the stay". The primary reason for folding two classes together is the difficulty in finding an identifier for both; however, it is also true that industry custom or practice sometimes dictates combination.

[19] Rockwell and McCarthy introduced conceptual congruency as an example of a specific implementation compromise. Such compromises were also discussed in McCarthy (1982), McCarthy and Rockwell (1989), Rockwell & McCarthy (1999), and Geerts and McCarthy (1997b)

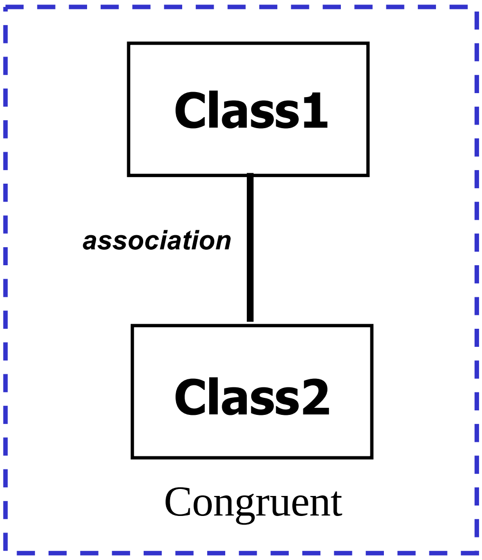


Figure 4-2 – Conceptual Congruency of Two Classes

In Figure 4-3, we have four common examples of simple REA conceptual congruency.

- In the northwest corner of 4-3, we have the case just mentioned where a single class fills the slot for both a commitment and an economic event. Besides the hotel stay example, other common examples include cases where a "purchase-order#" identifies both a purchase commitment and an actual purchase receipt or where a "production-order#" identifies both a planned production run and an actual production run. Both of these cases are common in industry practice, but technological advances – like using timestamps[20] here for the actual event instead of a document number—could undo the need for congruent classes in such cases.
- In the northeast corner of 4-3, we have the case of "nail" resources mentioned in chapter 3. This is a circumstance almost always brought about by representation constraints (i.e., identifier difficulties), although technological advances are clearly moving the identification possibilities frontier closer to obviating the need for congruency in many cases.
- In the southwest corner of 4-3, we have examples where a large-grained increment event (like a consulting job or an advertising campaign) becomes upon its completion the actual resource to be exchanged or consumed in a downstream revenue or production process. For example, completed audit engagements in effect become the economic resources that in-charge audit partners sell to their clients.
- In the southeast corner of 4-3, we have a congruency example common to simple cash exchanges, like a vendor selling a hot dog to a customer at a sporting event. In this case, the increment and decrement events occur almost simultaneously, so treating them as one occurrent makes representation sense.

[20] Lamport (1978) provides a good explanation of the use of timestamps to order all events.

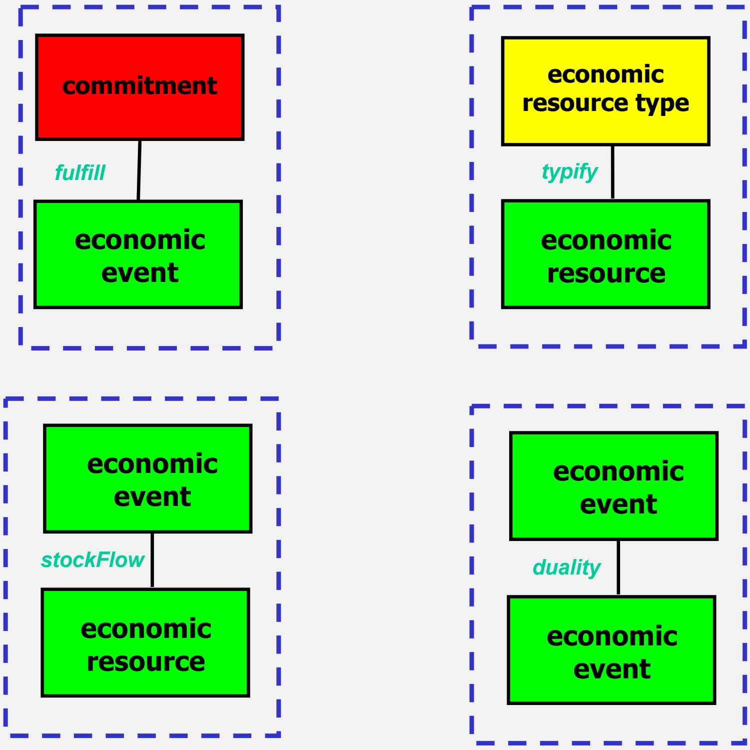


Figure 4-3 – Conceptual congruency examples (M2)

In Figure 4-4, we portray two more complicated examples of conceptual congruency, both of them dealing with duality associations:

- In the congruency example on the left (surrounded by dotted lines), we have three decrement economic events being combined that represent outflows of three different economic resources. Such combination is often necessitated by difficulty in finding an identifier for multiple outflows. For example, we could have a sale that uses inventory being combined with the truck being used to deliver the goods and with the labor of the truck driver doing the delivery. In this case we might have a single identifier (like a "timestamp" or an "invoice-number") for a class that represents all three with example attributes like "quantity of inventory sold," "miles driven by the truck," and "time spent by the driver."
- In the congruency example on the upper right (also surrounded by dotted lines), we have the quite common case of an increment event (like a cash receipt) being linked to an inflow resource (like cash), but which also uses some labor (like the time of the cashier) doing the event processing. Similar to above, we might have a single identifier (like a "remittance-advice-number" or a "timestamp") for the class with example attributes like "cash receipt amount" and "time spent by cashier."

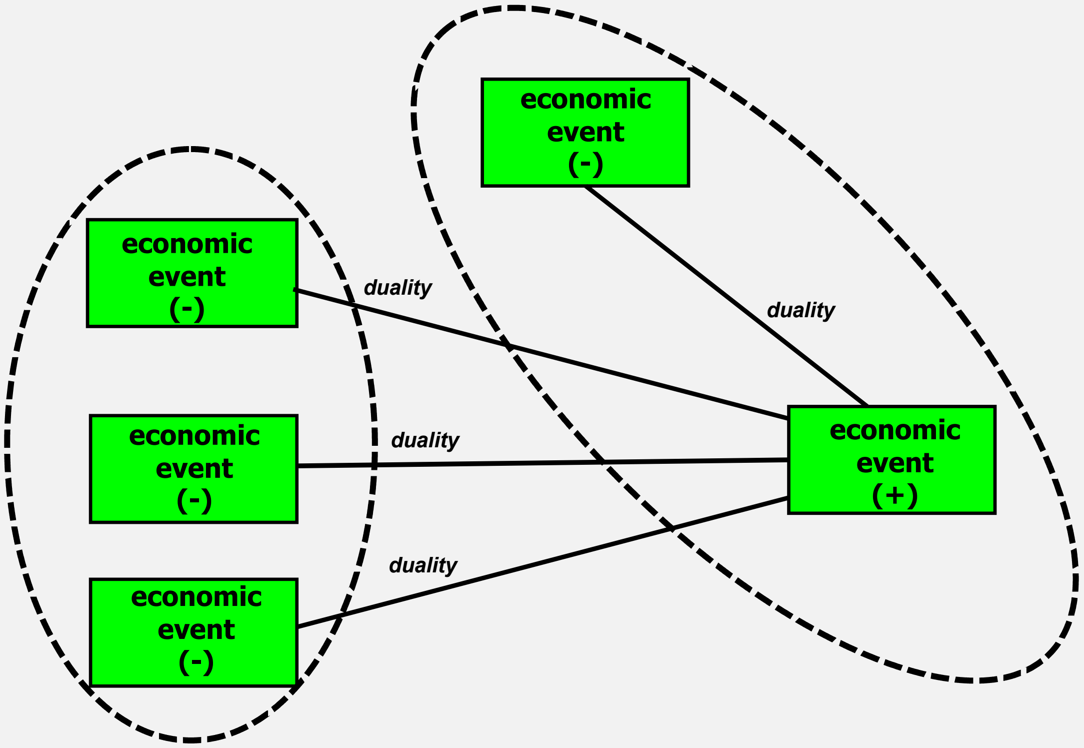


Figure 4-4 – Conceptual congruency of duality associations (M2)

In Figure 4-5, we show how conceptual congruency works with services like advertising service, insurance service, transportation service, etc. This is actually a congruency issue that was discussed in the original REA paper (McCarthy 1982 Figure 9(a)), and as a more particular example of a nail resource, it uses the same solution of materializing a class only at the type level. For the part of the diagram in Figure 4-5 surrounded by a horizontal circle, it is very difficult to find an identifier for a single instance of something like advertising service. The solution is to track it only at the type level with an identifier like "advertising service name" (with instances like *television advertising, newspaper advertising,* and *web advertising*). The typed resource could then use aggregated attributes like quantity of the resource available either in dollars or something like hours. These aggregated attributes summarize the effect of the economic events, "ServiceAcquisition" and "ServiceConsumption" shown in the vertical circle on the left of Figure 4-5, on the service. This would allow materialization of traditional accounting views like "advertising expense" and "prepaid advertising assets."

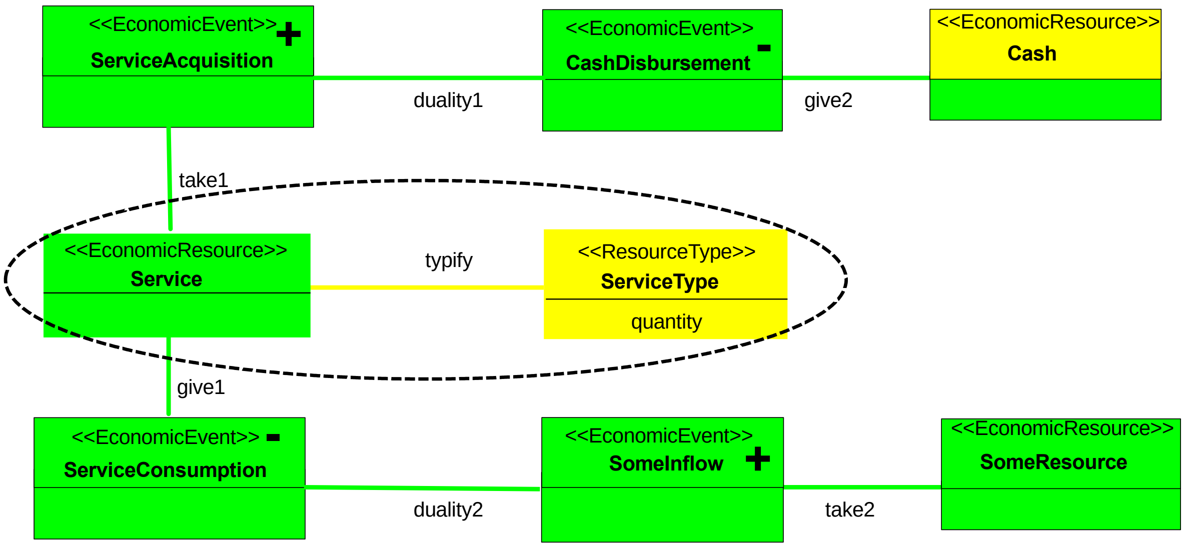


Figure 4-5 – Conceptual Congruency for Services Classes

In Figure 4-6, we illustrate our last example of conceptual congruency with the ubiquitous case of tracking employee service. The example shown illustrates (incompletely) the payroll business process on the left, while the right side portrays (again incompletely) the use of employee service in productive events like installing an engine in a car, driving a truck, conducting a purchase, etc. Again, the issue is the availability of an identifier to specify an instance of employee service – something which is almost impossible to supply consistently. The congruent solution here (surrounded by a vertical circle) is to simply view the stock and/or the consumption of available service as an attribute of the actual employee. Over the time of the payroll period, this stock and consumption would normally balance out, unless there was an idle time problem. In traditional manufacturing job costing, this balancing out is seen in the use of a "zero balance check" procedure that nets out time card hours (time spent on the premises) with job time card hours (time spent on individual jobs[21]). It is certainly the case that nearly all companies have similar payroll processes to that shown on the left of Figure 4-6. However, it is often the case that the consumption event on the right is not even materialized for many workers, in which case traditional accounting makes it a period cost. This is another area where source data automation technology is clearly moving the identification possibilities frontier closer to full REA modeling. For example, truck drivers now routinely have their time on various legs of their truck trips tracked by satellite.

[21] This is also the approach used by audit firms in the allocation of time to clients as billable hours.

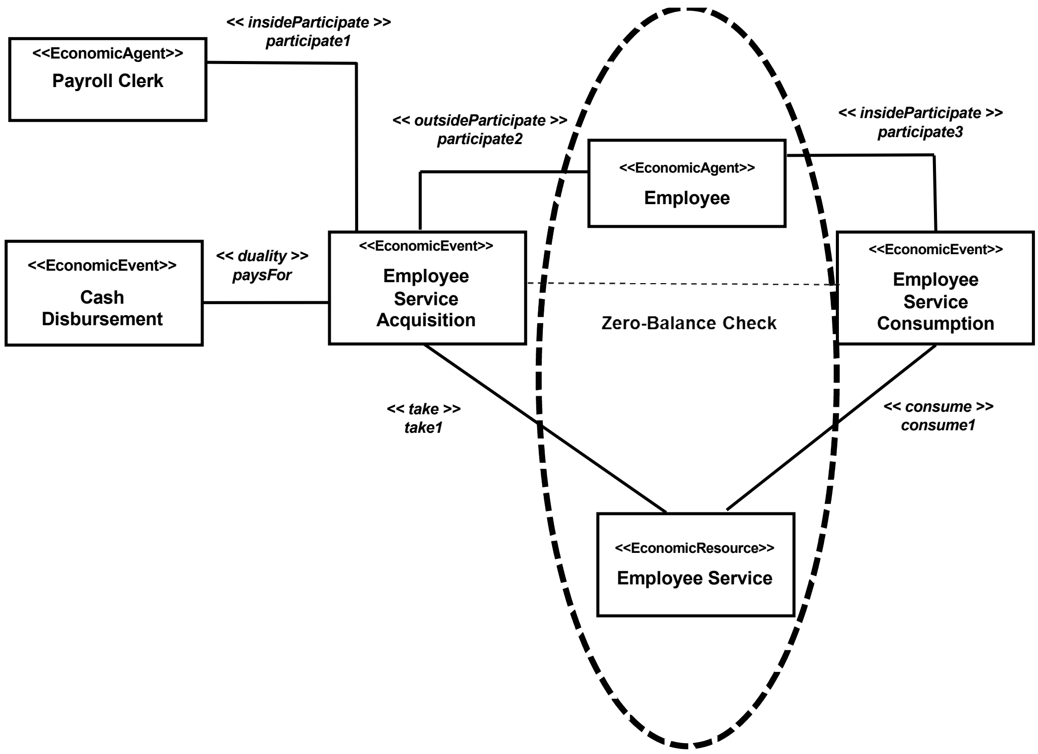


Figure 4-6 – Conceptual Congruency – Employee Service

Expansion of Metamodel Components – Meronymic Classes

Again, the REA metamodel of Figure 3-16 illustrates a business process frame with the individual classes representing slots in that frame. We demonstrated above how the notion of conceptual congruency can contract those slots. In this section, we show how those slots can be expanded. Such expansion is quite common when full design of a particular data representation needs to become more detailed. Basically, we treat the expansion as a whole-part constellation: an abstraction termed “meronymic inclusion” by Storey (1993, p. 463) who notes that the term derives from the Greek word “meros” for part. Storey also details seven different types of meronymic relationships, but we will concentrate here on just the higher level concept. More detailed analysis of meronymic expansion can be found in Winston et al. (1987), Motschnig-Pitrik (1993), and Gamallo (2013).

Figure 4-7 illustrates the basic idea of meronymic expansion – a representation of an example class (the whole) is expanded to include some of its component parts (the meronym). Examples of meronymic relationships include these:

- A finger is a meronym of a hand;
- An engine is a meronym of a car; and
- A song-performance is a meronym of a rock-n-roll concert.

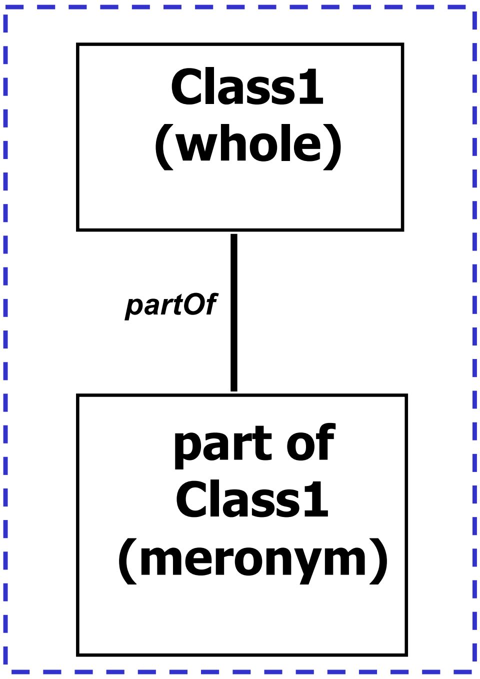


Figure 4-7 – Meronymic Expansion

Meronymic relationships are a subset of the more general "aggregation" abstraction pioneered by Smith and Smith (1977b) and explained in an accounting context by McCarthy (1987). Aggregation also applies when a business process is expanded to multiple subordinate business processes (see Figure 2-3 for example), when a single business process is decomposed into business events, or when one business event is decomposed into multiple business events. To aid in our discussion of different types of business processes (according to Coase (1937)), we portray in Figure 4-8 how aggregation relationships can be used to portray the REA granularity levels of Chapter 2. We use an augmented subset of that diagram next (Figure 4-9) to introduce the two basic types of process in REA: exchanges and conversions (Figure 4-10).

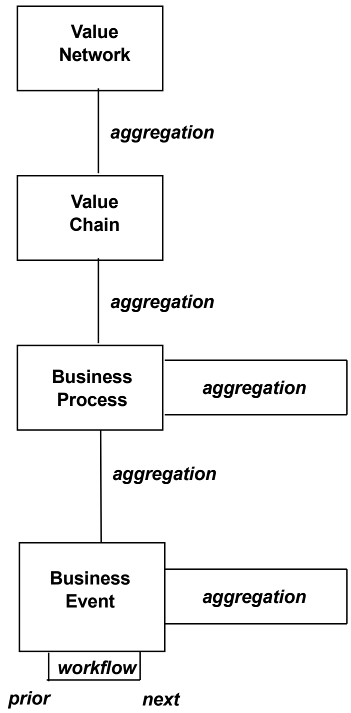


Figure 4-8 – Definition of REA granularity levels as aggregations

Accommodation of the Two Basic Coase Process Prototypes: Market Transfers (Exchanges) and Transformations (Conversions)

In this section we expand on these two process prototypes and provide far more detail on the classes and associations and the meronymic structures present in each.

In Figure 4-9, we re-examine two different types of notation (both introduced already) for portraying the interrelationships of an REA value chain with an REA business process. Figure 4-9(a) is derived (with some augmentation and reduction) from Figure 4-8, and in Figure 2-2 of chapter 2, we used the notation of 4-9(b) to illustrate an entire top-level value chain with nine business processes. Readers should understand how 4-9(b) is a shorthand notation for 4-9(a). In this section of the chapter, we analyze the overall nature of such business processes with a mind toward classifying them into two categories: exchanges and conversions.

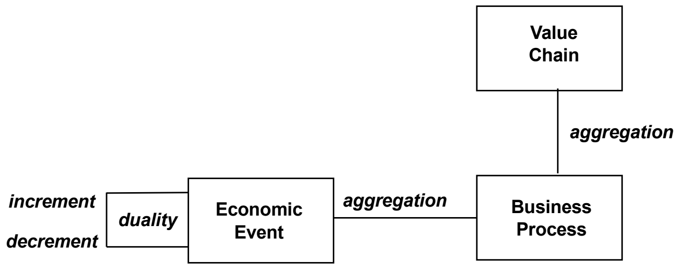


Figure 4-9(a) – Class Diagram of REA Granularity Levels 2-3

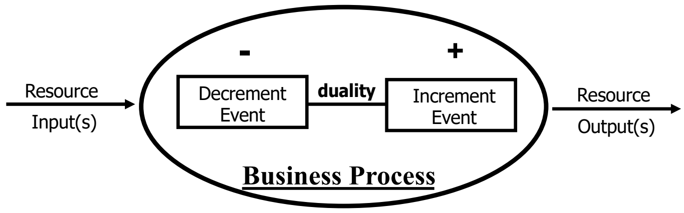


Figure 4-9(b) – Business Process Notation for Value Chains

Figure 4-9

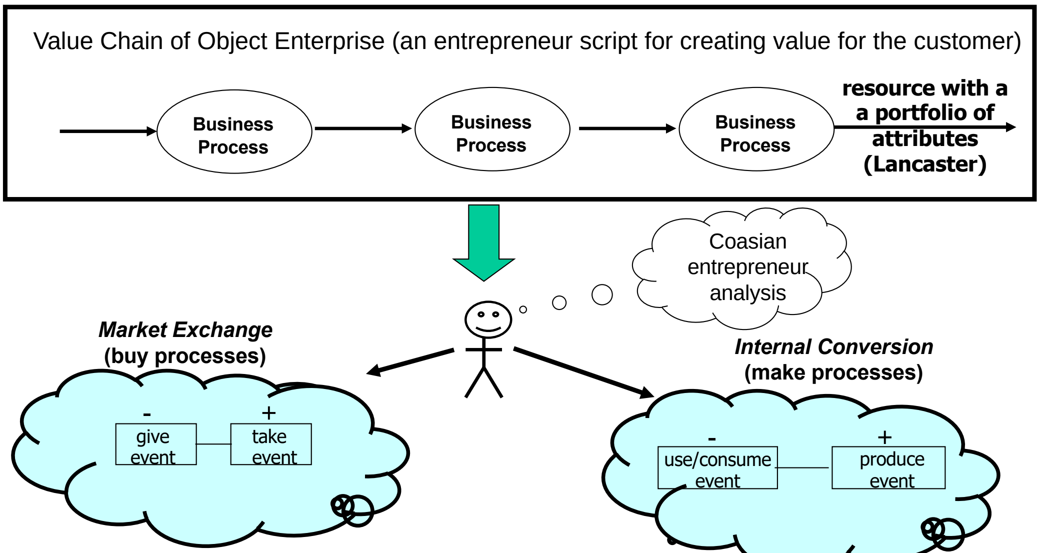
For REA, a value chain is considered an entrepreneur script for creating a final product with a portfolio of attributes of value to a customer (Geerts & McCarthy, 1997b; 1999; Lancaster, 1975) . In assembling each part of that portfolio, the entrepreneur must decide whether to "buy" that component or to 'make" that component (see Figure 4-10). This involves the classical entrepreneurial analysis pioneered by Coase (1937). The result of that analysis is that each node in the value chain becomes either (1) an arm's length market ***exchange*** using the price mechanism as a coordinating device between internal and external agents, or (2) an internal ***conversion*** coordinated by internal agents arranged in a multitier responsibility hierarchy (see chapter 1 appendix).[22]

According to Coase (1937) the entrepreneur decides whether to acquire the resource through a market exchange or an internal conversion based on a number of criteria. The overall decision to suppress the market's price mechanism might be done because it is more economical than executing a series of negotiations for the various means of production. A few of the issues to be considered when deciding to make the resource and eschew the market can include the costs of numerous transactions, the desire to limit the entrepreneurial attention to the process, and the firm's control over specific assets and specific human capital. Other factors involved in this decision are given at the bottom of Figure 4-10 and explained in more detail by Kroszner and Putterman (2009).

As shown in the middle of Figure 4-10 and expanded in Figure 4-11, this make/buy choice leads to slightly different REA models with two different types of dualities: ***transfer duality*** and ***transformation duality*** (Black & Black, 1929; Fisher, 1906). Transfers create value in a market transaction with outside parties, while transformations create value through changes in form or substance (Geerts & McCarthy, 2000b). Each of these variations is

[22] This choice between exchanges and conversion is discussed extensively in accounting textbooks, most commonly under the topic of "make or buy" decisions". See Horngren et al (2015) for accounting treatments of this topic.

explained in a separate section below. Next, we use a somewhat lengthy case, Alaskan Aircraft Expeditions (AAE) to further illustrate these concepts.



Transactions come to be organized within firms rather than in the market when the suppression of the price mechanism by the authority of the entrepreneur/coordinator is more economical than a series of exchanges negotiated between the relevant parties (Coase 1937). Conditions that affect this tradeoff include some of the following (Kroszner and Putterman, 2009):

- Transaction costs (like finding prices)
- Limited entrepreneurial attention
- Contractual arrangements and employment contracts
- Residual claims of equity and mixed financial structures

. Firm-specific assets and firm-specific human capital
. Transaction frequency and scale of production
. Costs of monitoring
. Lower powered incentives

Figure 4-10 – Entrepreneur Choice of Exchange vs Conversion

Example -- Alaskan Aircraft Expeditions (AAE)[23]

AAE generates revenues by offering various types of expeditions to its clients. Most clients avail themselves of a standard menu of expedition choices (such as "*The Glacier Bay Extravaganza*" or the "*Gold Rush Trek*", or the "*Bear Watch*."). Expeditions contain groups of 1-30 people.

In total, AAE offers a menu of 80 expedition types, each of which has a standard capacity, a fixed number of needed employees of certain types, one standard aircraft type, and a standard itinerary of places to be visited in a certain order. However, in some cases, a majority of the scheduled clients insist on an altered itinerary of their own making. In these cases (which occur on only 3% of the expeditions), AAE simply arranges a typed schedule (with typed fees, aircraft, and people) and then lets the lead guide accede to the client wishes, as long as the intended sites to be visited are in the location database. This database contains the names of all the "reachable" locations in Alaska, and it was populated by the company founders when they first got into the tour business. AAE tracks the locations actually visited by the individualized tours, but not those actually visited by the standardized tours (such tracking would be superfluous because standardized itineraries are *never* allowed to vary). The less common "tailored" expeditions are always typed into tour categories for management, resource planning, and staffing planning purposes.

[23] Alaskan Aircraft Expeditions is not a real enterprise, but it is a good extended example often used for REA teaching purposes. Its structure was designed from multiple observations of real Alaskan expedition enterprises.

Each client pays for his/her participation by obtaining an expedition ticket from AAE. Tickets may involve multiple passenger slots where for example there are associated slots for family members, friends, etc. An expedition ticket is issued by a booking agent to a single client for a single expedition tour, although there can be as many as 10 slots (taken passenger places) on the ticket (for a family with two parents and eight children for example).

A new scheduled expedition (identified by “expedition-number”) is usually put into the database months before its time by a booking agent, and the number of slots associated with it is determined by its type. Actual expeditions are keyed on a starting timestamp, and all actual expeditions are scheduled beforehand. Individual planes and workers are assigned to expeditions on the day they start, not beforehand. Expeditions have expedition tickets associated with them, and unique ticket numbers are generated automatically as they are needed. Per-slot charges to actual clients are often lower than their undiscounted basic fee which is determined by the type of expedition. Discounts are usually given for slots that are booked ahead of time, and the actual percentage for a discount is determined by a booking agent in negotiation with a client.

Expeditions involve one aircraft, 1-4 pilots, 1-3 guides, and 0-4 expedition workers plus 1-30 client and client-slotted associates. On all tours, one of the guides serves as the lead employee, responsible for management and safety policy. All guides may serve as leads. AAE aircraft are grouped by aircraft type, and the company uses 15-17 different types, each of which has a unique name. Individual aircraft (AAE usually has 50-100 leased at any particular time) are named after famous native Alaskans, and their seating and fuel capacities are determined by their type. Pilots are company-certified to fly only a subset of the aircraft types available, and their hours in each type are tracked.

Clients are also encouraged to buy expedition supply items (like hiking boots, fur parkas, or camping paraphernalia) at the company store on the starting day of their tour. AAE has a policy of associating "recommended" supply items for types of expeditions, and they distribute flyers to clients emphasizing their recommended buying lists. All client purchases of supply items are tied directly to a client's expedition ticket#. Supply items are never purchased directly by clients because a local commercial statute prohibits AAE from direct sales without an impending participation on an expedition. Supply items are manufactured directly by AAE on the grounds of their own facility (more details of the manufacturing process are supplied later in this chapter).

Clients may pay for expedition tickets in cash taken directly by the booking agent. Clients may also pay by credit card, and they may keep multiple cards in their file. All cash receipts are keyed on a remittance timestamp, even when money is taken directly. It is often the case that clients pay for their initial expedition ticket fee right away, then pay for their remaining balance due to supply items a month or two later, because no money or credit cards can be used in the company supply store (items purchased are charged to a ticket). Clients often do buy multiple expedition tickets at one time (if for example, they want to go on two different expeditions on consecutive days). When clients buy multiple tickets, they may settle with a single remittance. For AAE, 90% of cash receipts occur in the revenue cycle.

AAE uses debt financing, and it does most of its acquisitions on credit where terms vary by vendors. The company leases its aircraft, and it also leases its structures (an office building, a small factory, and an aircraft facility). Advertising costs are tied to expedition types and and/or customer segments, and structure costs are tied to temporal/sectional groupings of company activities.[24]

This next section focuses on the top half (starred) portion of the Figure 4-11 by examining Alaskan Aircraft Expeditions' revenue process. Readers should note the names for the increment event (take) and decrement event (give) are different than those for conversions.

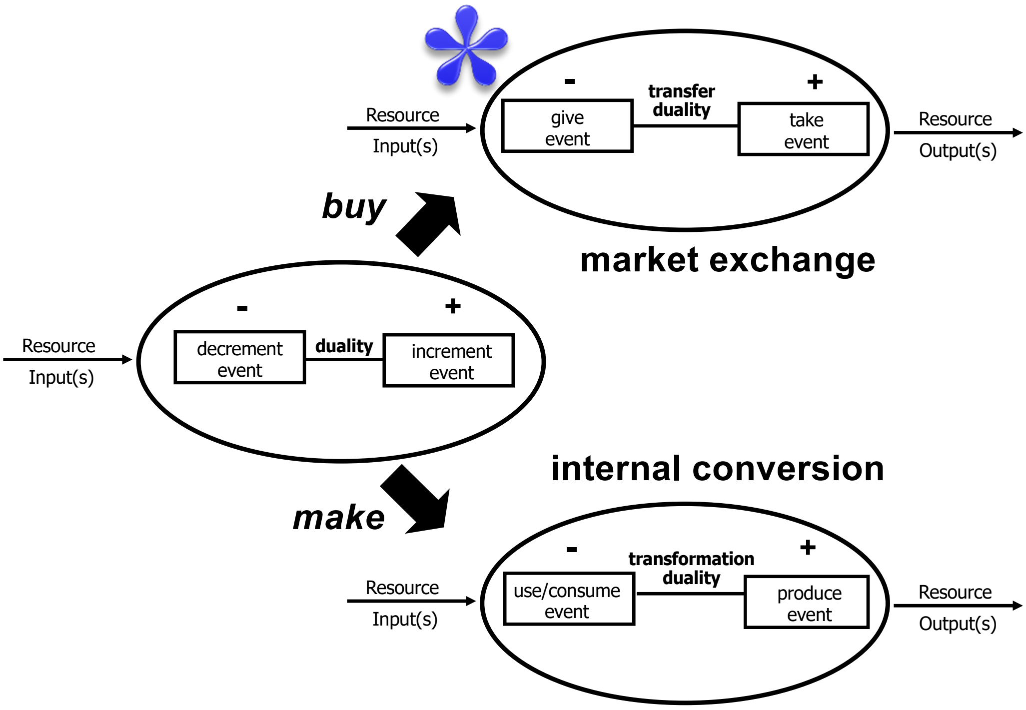


Figure 4-11 – REA duality types: buy (transfer) vs. make (transformation)

REA Market Exchanges

Our example company – *Alaskan Aircraft Expeditions (AAE)* -- conducts tours with aircraft to various locations, so its revenue process is one where they use an aircraft and consume varying amounts of employee labor on expeditions. "Expedition" is thus the decrement event in the example process (see Figure 4-12). However, the whole expedition is broken into "expedition tickets" for the purpose of matching customers to a particular tour, and what they pay for is a ticket (obtained from a booking agent) that allows them to be part of the expedition. "Expedition Ticket" is a meronym of "Expedition." In Figure 4-13, we illustrate this in more detail as a process level REA class diagram with the whole-part constellation again surrounded by a dotted line. Readers should note how expanding this economic event into parts allows better specification of the relationships connecting the event to its resources and agents.

[24] Allocation of facility costs will not be discussed in this chapter, but in chapter 6.

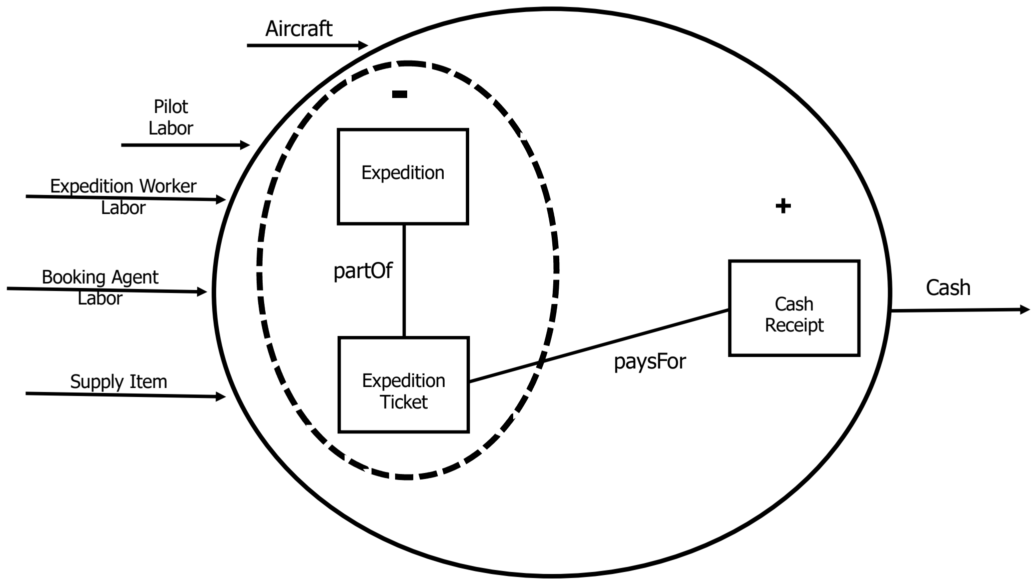


Figure 4-12 – AAE Value Chain Process with Meronymic Expansion of an Event Slot

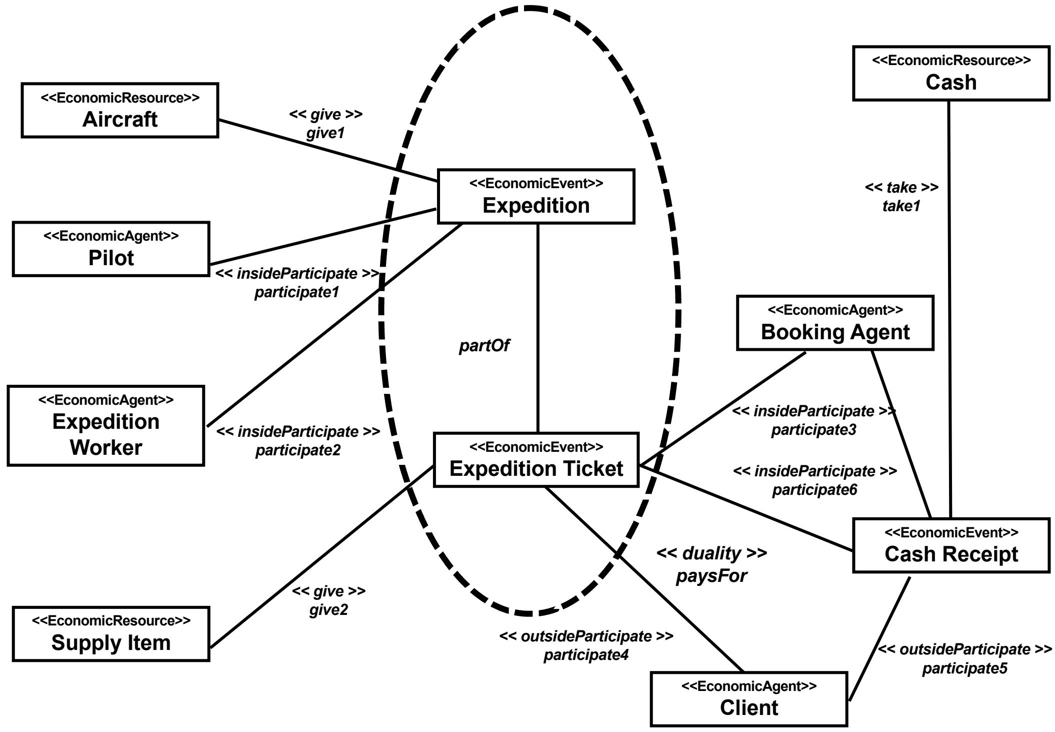


Figure 4-13 – Meronymic Expansion of an Event Slot

The REA accountability pattern for a market exchange is shown in Figure 4-14, and it mirrors the REA model (M2) previously shown in Figure 1-5a. An example M1 model of an exchange (a raw material acquisition process) was illustrated in considerable detail in Chapter 1. The exchange and the ultimate value accrued in the process is coordinated by the price mechanism as it operates between two independent parties.

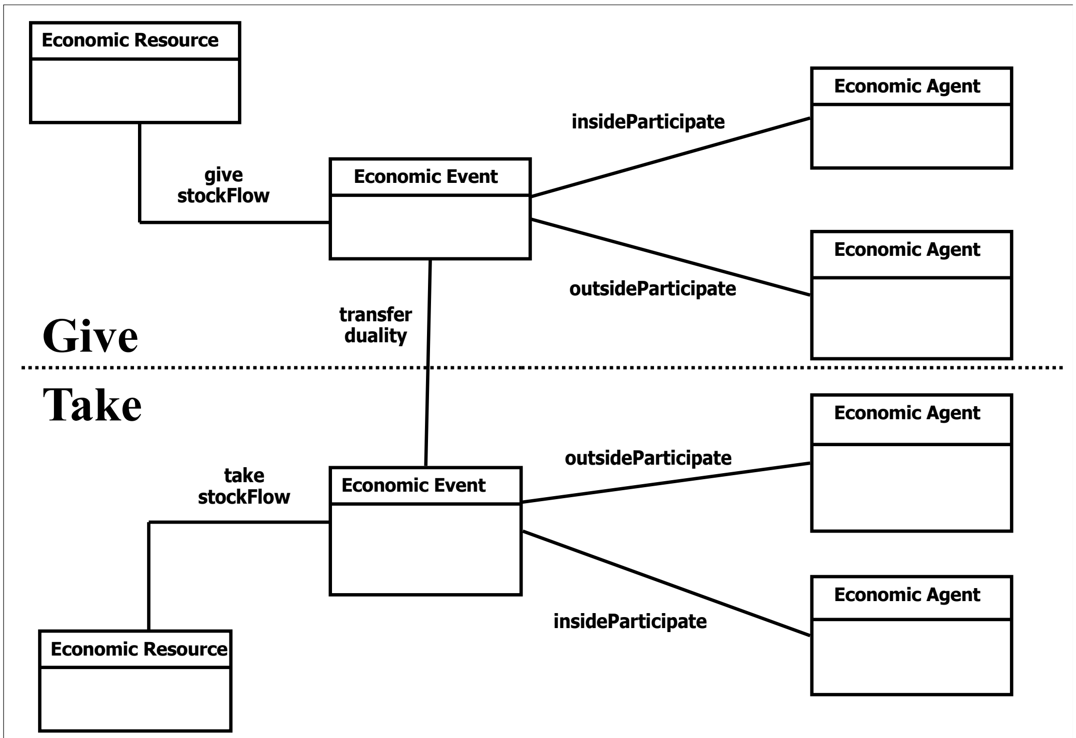


Figure 4-14 – Accountability Pattern for Market Exchange

In Chapter 2 and in Figure 4-8 above, we illustrated that an REA business process could actually be conceived as an aggregation of multiple smaller activities which REA terms ***business events*** arranged in a particular order called a ***workflow***. In Figure 4-15, we introduce the additional REA concept of ***business process phases***, an innovation adopted directly from the Open-edi standard of ISO 15944-1 (2001). This expands our aggregation hierarchy slightly as it now illustrates that a *business process* is an aggregation of ***business process phases*** which themselves in turn are aggregations of ***business events*** arranged in a workflow. As illustrated in Figure 4-16, our five process phases will be termed: planning, identification, negotiation, actualization, and post-actualization.

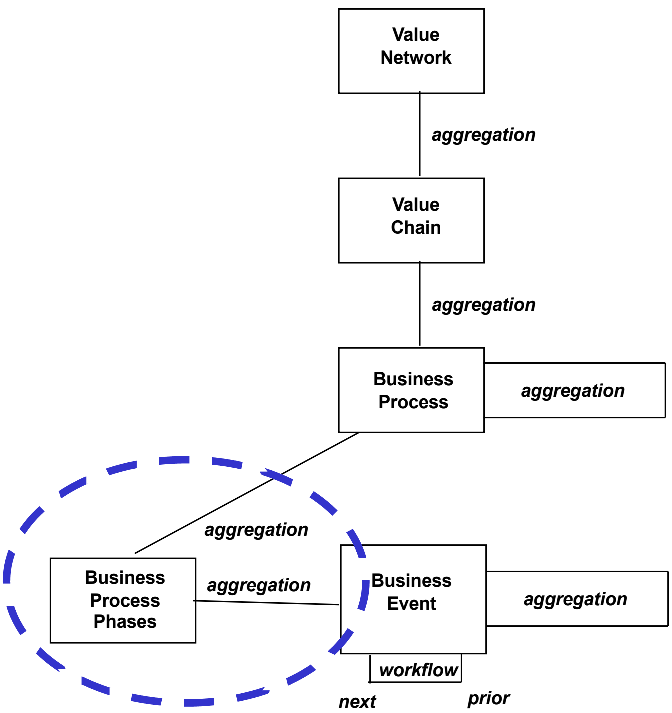


Figure 4-15 – Adding Process Phases to REA Granularity Levels

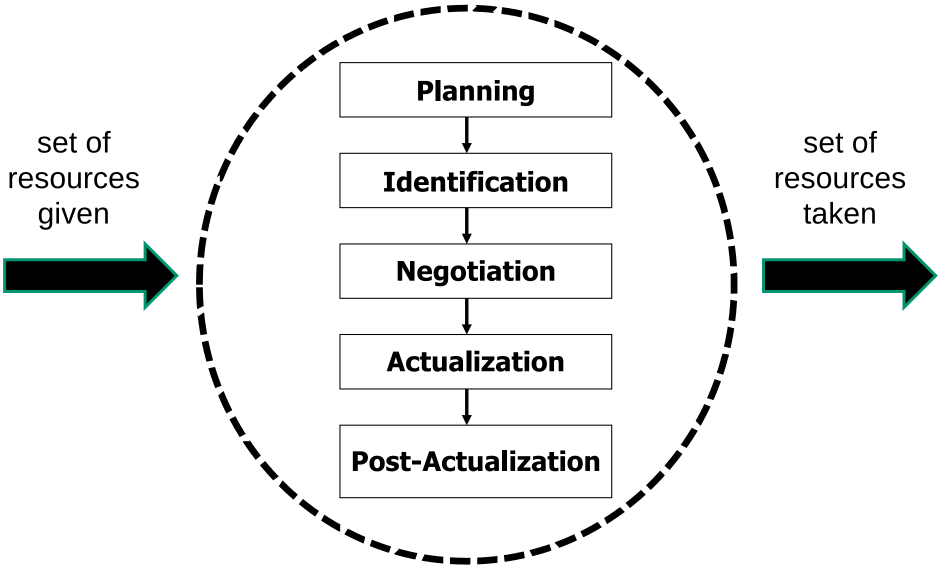


Figure 4-16 – Market Exchange Process Phases

As explained in ISO 15944-1, the editors of this standard[25] derived these five phases from an exhaustive examination and compilation of the management literature dealing with business process engineering (see Annex F of ISO 15944-1 (2001)). Their analysis was then adopted into part 4 of the Open-edi standard series (ISO 2007) which relies heavily on the REA ontology. The ISO description of the phases is quoted exactly in Figure 4-17. With regard to REA, ***buyer*** and ***seller*** are obviously roles that ***economic agents*** can play, and an

[25] Two advanced Canadian IT practitioners: Jake Knoppers and David Clemis.

Open-edi ***business transaction***[26] is a term synonymous with the REA concept of ***business process*.**

- Planning: In the Planning Phase, both the buyer and seller are engaged in activities to decide what action to take for acquiring or selling a good, service, and/or right.
- Identification: The Identification Phase pertains to all those actions or events whereby data is interchanged among potential buyers and sellers in order to establish a one-to-one linkage.
- Negotiation: The Negotiation Phase pertains to all those actions and events involving the exchange of information following the Identification Phase where a potential buyer and seller have (1) identified the nature of good(s) and/or service(s) to be provided; and, (2) identified each other at a level of certainty. The process of negotiation is directed at achieving an explicit, mutually understood, and agreed upon goal of a business collaboration and associated terms and conditions. This may include such things as the detailed specification of the good, service, and/or right, quantity, pricing, after sales servicing, delivery requirements, financing, use of agents and/or third parties, etc.
- Actualization: The Actualization Phase pertains to all activities or events necessary for the execution of the results of the negotiation for an actual business transaction. Normally the seller produces or assembles the goods, starts providing the services, prepares and completes the delivery of good, service, and/or right, etc., to the buyer as agreed according to the terms and conditions agreed upon at the termination of the Negotiation Phase. Likewise, the buyer begins the transfer of acceptable equivalent value, usually in money, to the seller providing the good, service, and/or right.
- Post-Actualization: The Post-Actualization Phase includes all of the activities or events and associated exchanges of information that occur between the buyer and the seller after the agreed upon good, service, and/or right is deemed to have been delivered. These can be activities pertaining to warranty coverage, service after sales, post-sales financing such as monthly payments or other financial arrangements, consumer complaint handling and redress or some general post-actualization relationships between buyer and seller.

SOURCE: ISO FDIS 15944-1 – Operational Aspects of Open-edi for implementation

Figure 4-17 – ISO Open-edi Phases of an Exchange

Introducing phases as an REA component allows us to be more specific about the detailed nature and organization of business events, and the use of these phases is also extremely useful in crafting easier explanations of very complicated business processes. The exact delineation and ordering of the phases is slightly arbitrary at the margin, but they can be usefully exercised to order very complex and very long sequences of workflow events. For REA, we will assume for the most part that they occur in the order shown and that they categorize business events. These assumptions will make the phases useful for state machine mechanics[27], an advanced REA technique to be described in Chapter 6.

In Figure 4-18, we illustrate how the exchange business process phases correspond (in very loose fashion) to components of the REA ontology:

- In planning and identification phases, the two agents engage in activities (REA business events) to determine the types of resources they plan to acquire in typed events (like wholesale vs. retail exchanges). They then determine the types of trading partners they need and ultimately identify one particular partner.
- In negotiation, the trading partners use additional business events to ***specify*** the typed conditions for commitments and contracts.

[26] *Business transaction* is a fundamental construct of the *Open-edi Reference Model* (International Standards Organization, 2010).
[27] REA state machine mechanics are described in ISO 15944-4 (2007).

- In actualization and post-actualization, the inside and outside agents engage in business events (workflow activities) that ***fulfill*** the exchange specifications. When a business event completes a transfer from one party to another, an economic event is registered.[28]

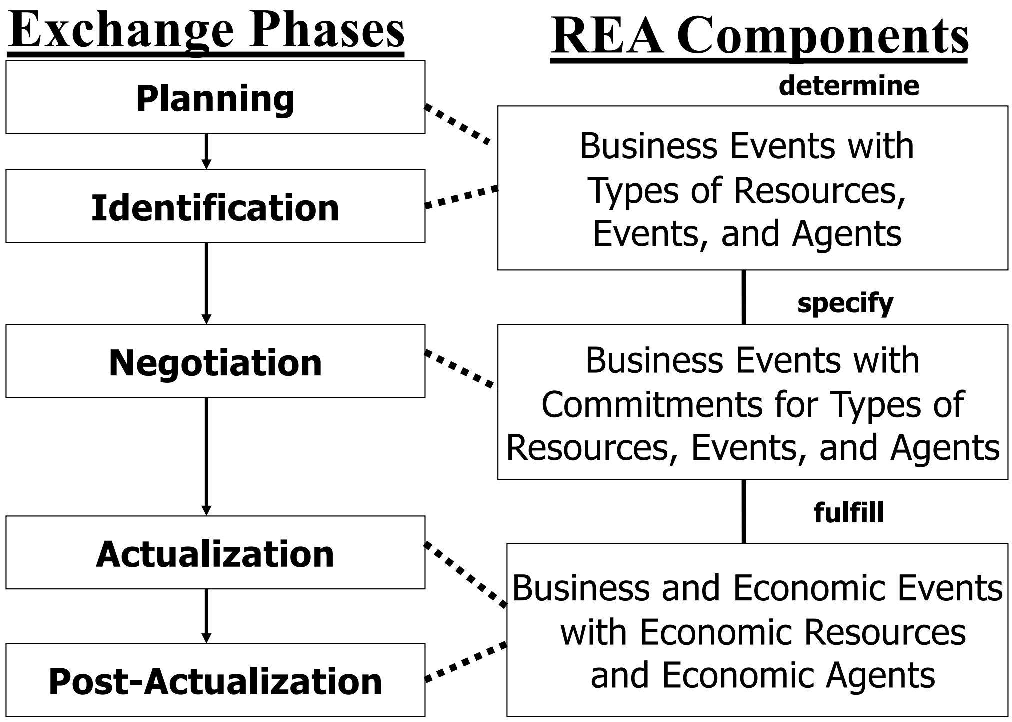


Figure 4-18 – Correspondence of Exchange Phases and REA Components

In Figure 4-19, we illustrate a business process phase example at the M1 level with 14 business events.[29] Again, this particular illustration example is based on our acquisition example from earlier chapters, with the actualization business events corresponding exactly to the workflow shown in Figure 2-4. The exchanged electronic messages (shown in italics in Figure 4-19) have actually been adopted from EDI (Electronic Data Interchange) standards like X12 or EDIFACT,[30] but readers may imagine that the same **determine-specify-fulfill** dialogue could have been accomplished by paper documents or by speech acts in a less technically advanced environment (such as a buyer and seller negotiating across a table).

[28] A business event that signals the completion of an economic event is often referred to as a ***critical*** event. The values represented in accounting financial statements are summaries of events in the actualization and post-actualization phases.
[29] This example has been approximately adapted from ISO 15944-4 (2007).
[30] EDIFACT is the United Nations EDI standard, while X12is an alternative EDI standard used in the USA.

Business Process Phase		Example Business Events
Planning	1	Buyer sends *CatalogRequest* to Vendor
	2	Buyer receives *Catalog* from Vendor
Identification	3	Buyer sends *AvailabilityAndPriceRequest* to Vendor
	4	Buyer receives *AvailabilityAndPriceResult* from Vendor
Negotiation	5	Buyer sends *Offer* to Vendor
	6	Buyer receives *CounterOffer* from Vendor
	7	Buyer accepts the details of *CounterOffer* on shipment and proposes *PaymentSchedule* to Vendor
	8	Vendor accepts *PaymentSchedule* from Buyer, thus completing *PurchaseOrder* (contract) specification (alternatively, another *CounterOffer* would loop negotiations or a *NonAcceptance* would suspend or abandon the business process)
Actualization	9	Receiving Clerk receives an *AdvanceShippingNotice* from *Vendor* when goods are shipped
	10	Receiving Clerk receives and inspects Raw Materials
	11	Receiving Clerk sends *ReceivingReport* to Vendor and Buyer when inspected goods are accepted (Economic Event *Purchase* registers)
	12	Cashier receives *Invoice* from Vendor
	13	Cashier sends *BankTransferNotice* to Vendor with information about payment of the *Invoice* (Economic Event *Cash Disbursement* registers)
Post-Actualization	14	Buyer sends *WarrantyInvocation* to Vendor if materials are found to be inadequate in use

Figure 4-19 – An Example Exchange Process with Phases

In Figure 4-20, we portray the important alternative of exception processing in business process activities. In object-oriented development, the sequence of events where all activities go as planned is termed the "happy path." When problems occur (such as inability to negotiate a price, abandoned negotiations or sales returns), the sequence is an "unhappy or exception path." Contracts commonly indicate several possible exception paths with differing economic events for the actualization and post-actualization phases of a business process, while exception conditions are usually less formally determined for earlier phases. In Figure 4-20, we illustrate several exception (i.e., unhappy) examples, but for simplicity in this monograph, we will usually limit ourselves to the happy path shown on the left where the phases unfold without exception from start to finish.

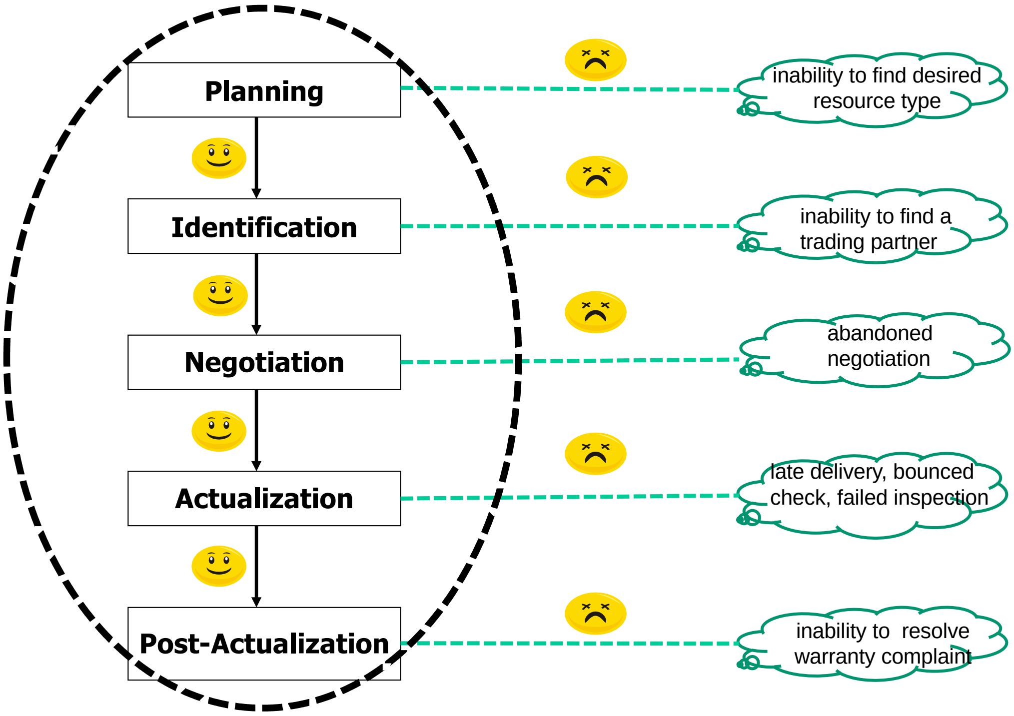


Figure 4-20 – Happy and Unhappy Business Process Paths

A preliminary value chain for AAE is shown in Figure 4-21.[31] We have labeled each business process at this top level with the common cycle description normally used in accounting. In Figure 4-22, we illustrate in some degree of detail[32] what the M1 model of the AAE revenue process might look like. We show the stereotypes for the REA-based classes, but for economy of space purposes, we do not show the stereotypes for associations. Instead, we label the associations with variants of their stereotyped names. The detailed exchange example of Figure 4-22 illustrates many of the REA-based principles we have described in our first four chapters: REA classes including economic resources, events, and agents, the type level of these classes, and REA policies, commitments, and accountability layers. At this point, readers should also note that realistic modelling of business processes will usually include both classes and associations beyond the REA metamodel components.

[31] This value chain is an expansion of figure 2-2, but it has also been deliberately simplified for explanation purposes.
[32] We have chosen the amount of detail carefully in an effort not to make the figure lettering too small. Obviously, a real data model would have much more detail.

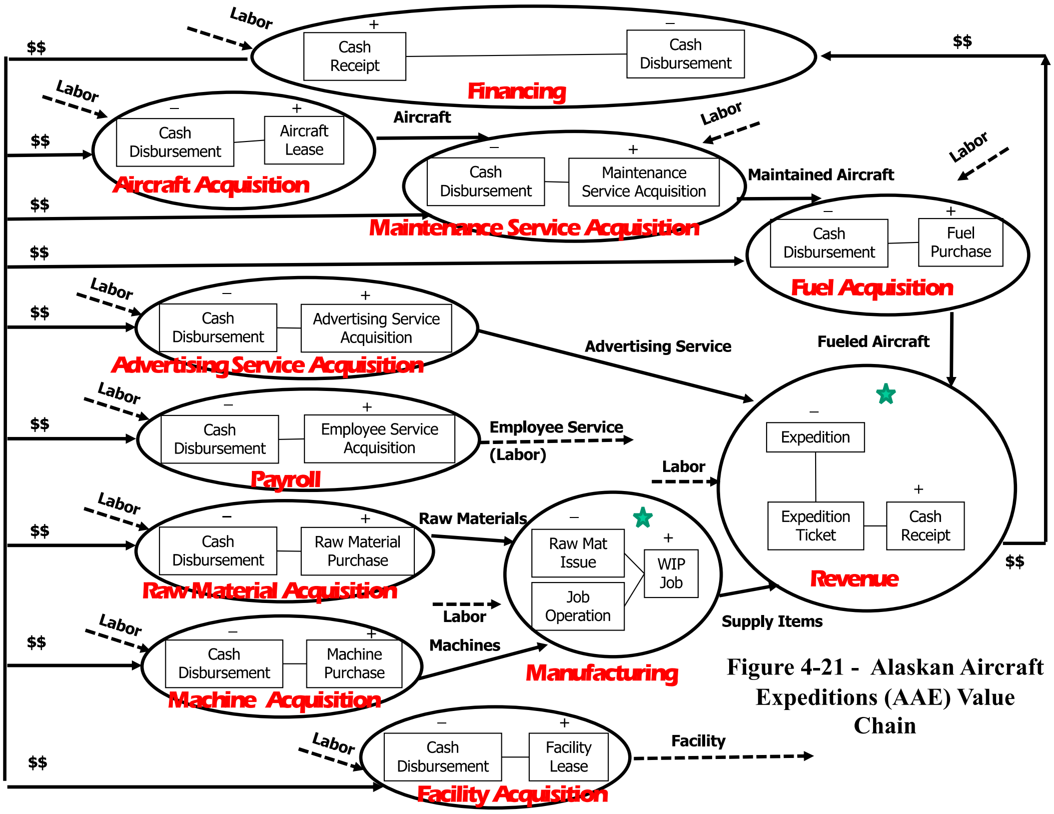


Figure 4-21 - Alaskan Aircraft Expeditions (AAE) Value Chain

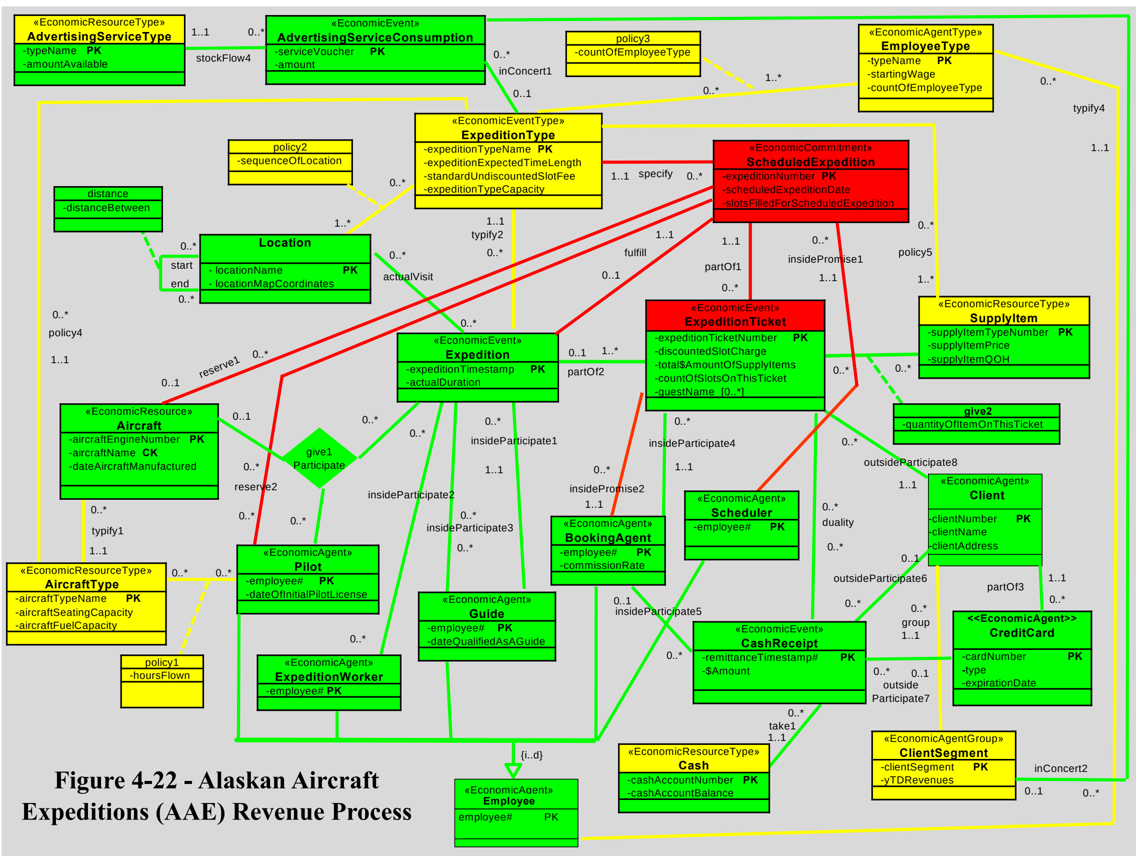


Figure 4-22 - Alaskan Aircraft Expeditions (AAE) Revenue Process

In the next section, we augment this analysis by describing the REA modeling of **internal conversion** business processes (see Figure 4-23).

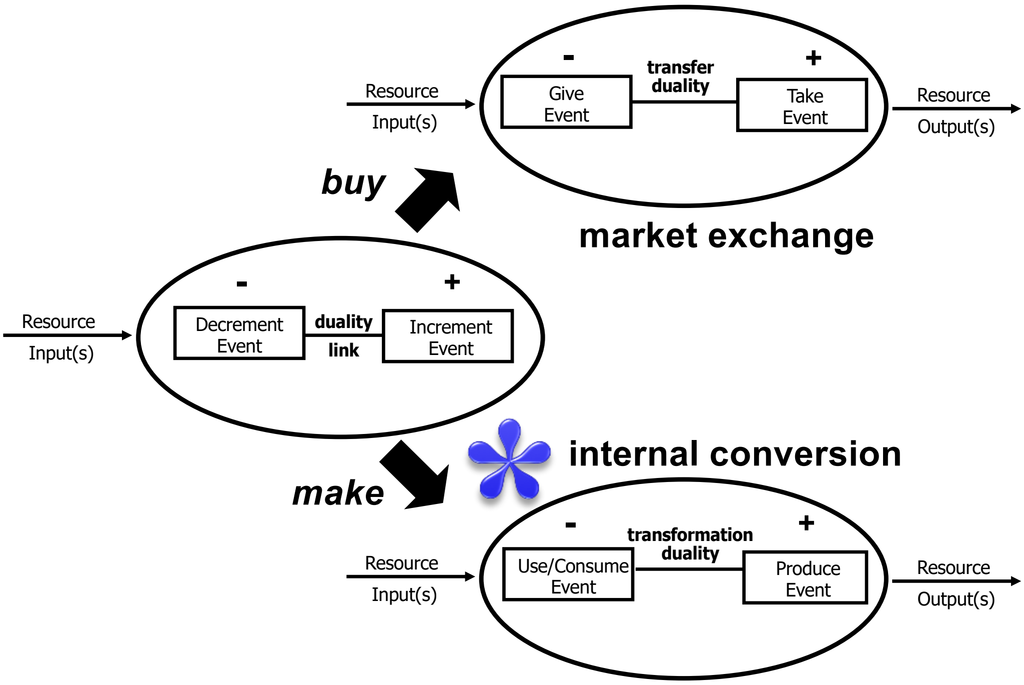


Figure 4-23 – REA duality types: buy (transfer) vs. make (transformation)

REA Internal Conversions

The nature of conversion process modeling has developed throughout the history of REA design work. In the original REA paper (McCarthy 1982) and in follow-up proofs of concept (Fedorowicz & McCarthy, 1983), conversions in general and manufacturing in particular were viewed as a series of arms-length exchanges from one department or agent in a company to another – a common transfer pricing conception. Beginning with workshop presentations by Geerts and McCarthy (1992a; 1992b), new views of REA manufacturing began to surface (Geerts & McCarthy, 1994). These evolved over six years to Geerts and McCarthy (2000) then further to our present work as Fisher's (1906) distinction between multiple types of duality interactions (exchanges vs. transformations) received greater emphasis.[33] This evolution in REA thought was aided by multiple factors:

- Consultation with industry experts in manufacturing planning systems and supply chain management like Robert Haugen (Haugen & McCarthy, 2001a; 2001b). Especially prominent concepts, advocated by Haugen, were accommodation of bills-of-materials modeling with quantity-per attributes for capacity management and plant scheduling.
- Review of REA modeling papers that emphasized complex conversion process examples (see Grabski and March (1994) and Denna et al. (1994)).The examples there were much more complex than those in the prior REA literature, and accommodation of this increased complexity led to basic changes in the pattern for conversions.
- Reorientation of REA manufacturing examples in teaching that utilized the increased complexity mentioned above. Examples include *NicoSys* (McCarthy W. E., 1999),

[33] In actuality, Fisher specified a third type of dual interaction called transportation. In this present work, we consider – in accordance with the economic resource ideas of Lancaster (1966) – that transportation is a subset of transformation where the location attribute is changed.

Bismark Sausage (McCarthy W. E., 2000), *Western Michigan Office Furniture* (McCarthy W. E., 2005), and *Nantasket* (McCarthy W. E., 2006).

All of this experience led to the following changes in our present approach to differential modeling of conversions vs. exchanges:

- Reconsideration from the theory of the firm perspective (Kroszner and Putterman 2009) that the arms-length idea of transfer pricing with two agents (inside, outside) was not as suitable from an economic theory perspective as is modeling internal agents only, as arranged in a tiered responsibility hierarchy (see Appendix of Chapter 1) where planning and coordination are less dependent on price mechanisms (see Figure 4-10).
- Reconsideration of the philosophical basis for modeling the accounting concept of *Work-in-Process* (WIP) where Geerts and McCarthy (1999) realized that the traditional accounting concept of a WIP balance is simply an attribute of an interrupted economic event of significant time length like a job, a batch, a manufacturing run, or a campaign. From an ontological perspective (Sowa 2000), all of these entities are best treated as processes (occurrents), not things (continuants).
- Realization that the give (use or consume) and take (produce) nature of ***transformation duality*** (Geerts & McCarthy, 2000b) was meaningfully different enough to warrant differential modeling from ***exchange duality***, most significantly in its meronymic nature.

These changes lead to the conversion process pattern illustrated in Figure 4-24. Readers should note how this pattern differs from the exchange pattern of Figure 4-14, especially with regard to its designation of one (inside) agent only and its differential labeling of the *stockFlow* labels. ***Use*** is an outflow that does not completely exhaust an input factor (like a machine being used in a job operation), while ***consume*** is an outflow that does exhaust the input (like raw material being consumed in a raw material issue). ***Produce*** is an inflow that links an overall assembly process to its finished product.

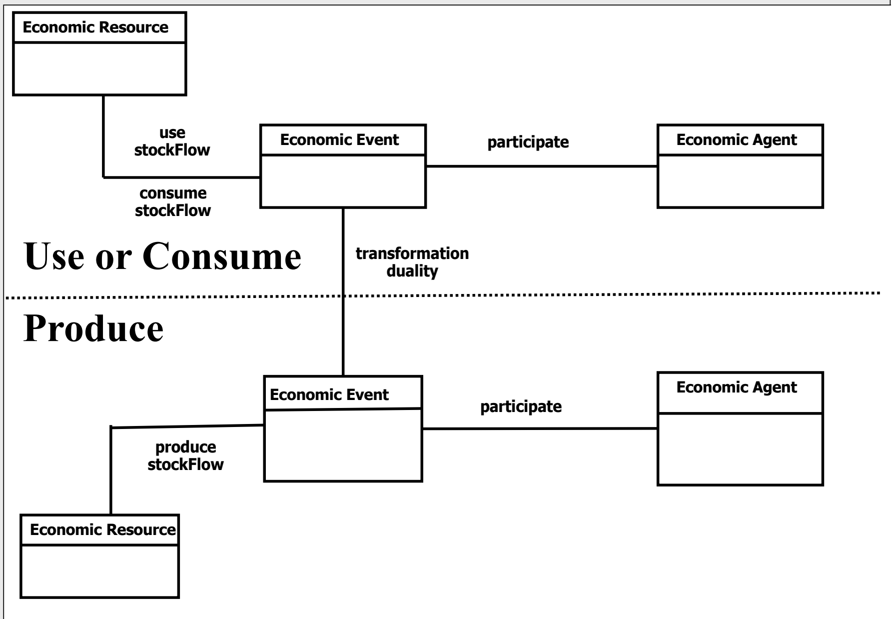


Figure 4-24 – Accountability Pattern for Internal Conversion

In Figure 4-25, the meronymic nature of conversion processes is emphasized. On the left, the **parts** of a conversion process are portrayed: relatively short economic events (finer granularity) that use or consume resources like “issue an engine to an automobile in process” (an instance of a *raw material issue*) or “install an engine in an automobile in process” (an instance of a *job operation*). On the right, the **whole** conversion process is portrayed: an economic event of longer duration (coarser granularity) that produces the expected outcome of the conversion like “make a batch of Cadillacs over the course of three days” (an instance of a manufacturing run).

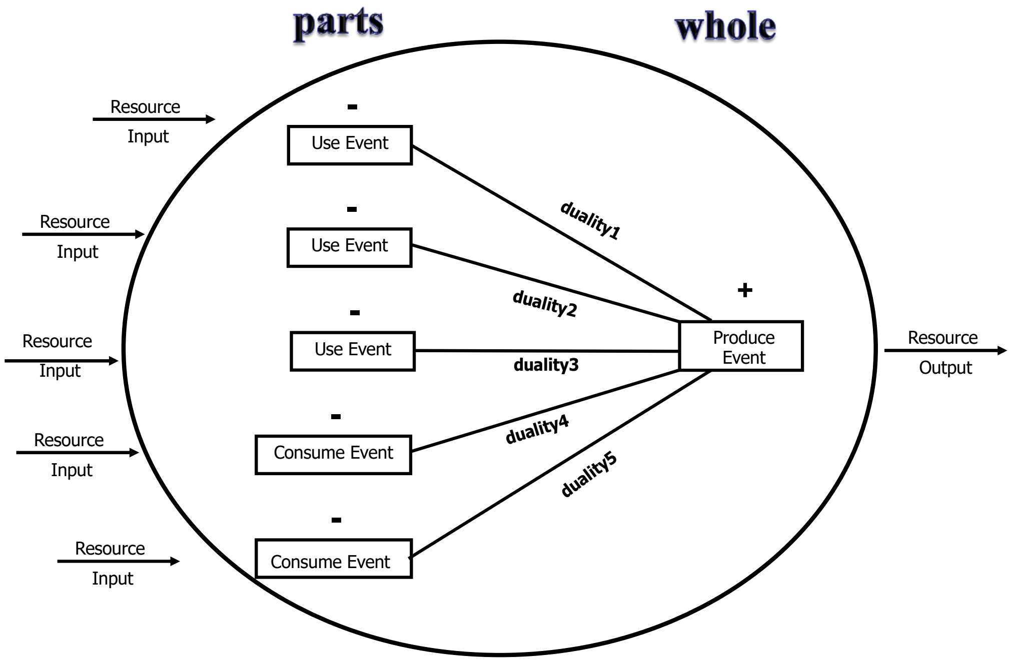


Figure 4-25 – REA Transformation Duality (Meronymic)

In Figure 4-26, Figure 4-27, and Figure 4-28, we illustrate how the phases of a business process introduced earlier need to be adapted for a transformation. Those five phases – planning, identification, negotiation, actualization, and post actualization – were adopted from the world of Open-edi (ISO 15944-1) where activities with business and economic events take place primarily in the collaboration space between independent firms.

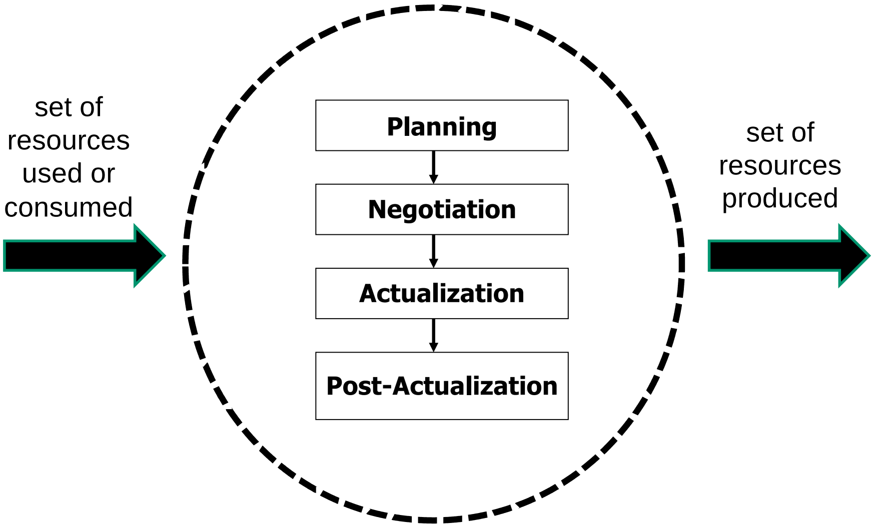


Figure 4-26 – Conversion Process Phases

By contrast with exchanges, conversions occur within just one firm. As Figure 4-26 illustrates, this means that the identification phase of a business process (where a 1-to-1 linkage with an independent outside agent is established) is no longer needed. In Figure 4-27, we provide short textual descriptions of the conversion phases. Readers are reminded that these categorizations are not absolute. The categories provide useful delineation of business and economic events, and we normatively expect the conversion workflow to follow these phases. In Figure 4-28, we illustrate how these phases might unfold for Alaskan Aircraft Expeditions as they manufacture the supply items for their company store.

- Planning: In the Planning Phase, economic agents are engaged in activities to decide the types of actions to take for converting certain factors of production into a higher order bundle of economic resources – the decision to make a resource and the determination of the types of activities needed for that conversion process.
- Negotiation: The Negotiation Phase pertains to all those actions and events where potential economic agents have (1) selected the specified type and quantity of economic resources to be used or transformed and, (2) scheduled the detailed activities needed to produce the specified resource set. For an internal conversion, the detailed set of types of resources and types of agents needed for the entire conversion process are converted from a quantity-per basis to actual scheduled amounts and quantities.
- Actualization: The Actualization Phase pertains to all economic and business events necessary for the execution of the results of the scheduled business process. The firm assembles the planned combinations of materials, labor, and other services into the bundled package of a finished resource or a completed process.
- Post-Actualization: The Post-Actualization Phase includes all of the events and associated exchanges of information that occur after the agreed upon good, service, and/or right is deemed to have been manufactured or assembled. These can be activities pertaining to some rescheduling/adjustment of the actual or proposed standards for the conversion processes.

Figure 4-27 – ISO Open-edi Phases of a Conversion Process

Business Process Phase		Example Business Events
Planning	1	Production supervisor receives sales forecast for types of supply items (finished goods)
	2	Supervisor evaluates capacities and available sources of skilled labor (employee types), machines (machine types) and raw materials (raw material type)
Negotiation	3	Using the bill of materials, and the routing list , the production supervisor uses typed quantity-per attributes to derive scheduled events and attributes.
	4	Supervisor schedules a production order (commitment) with reciprocal links to scheduled RM issues (commitments) and to scheduled labor operations (commitments).
Actualization	5	Factory workers perform raw material issues and job operations until the entire production job is completed, thus making finished goods or other completed processes available.
Post-Actualization	6	Production supervisor reschedules some capacities and resources based on breakages/shortages/overages.

Figure 4-28 – An Example Conversion Process with Phases

Alaskan Aircraft Expeditions Conversion Cycle

In Figure 4-29 we again illustrate in some degree of detail what an M1 model of an AAE process might look like, this time for its manufacturing. Again, we show the stereotypes for the REA-based classes but only label the associations with variants of their stereotyped names. This UML class diagram possesses ample detail for a reader to peruse, but these features might be especially notable:

- The three REA temporal levels – policy-scheduling-accountability – as indicated by the yellow, red, and green colors – are discernible from top to middle to bottom of the figure. This is an M1 model of a particular REA application, and readers should note its correspondence to the M2 REA metamodel portrayed at the end of the last chapter (Figure 3-16).
- Some attributes exist at all three temporal levels. As an illustration, there is a standard sequence for operations, a scheduled sequence, and then an actual sequence. Under certain circumstances, these could all differ for good business reasons. It should be noted also that these three levels of temporal delineation supply a conceptual framework for the attributes needed to calculate normal cost accounting manufacturing variances (as outlined in a text like (Horngren, Datar and Rajan 2015)).
- The diagram is somewhat incomplete in that it doesn't account for machines as an input (as shown on the value chain). This is due to space considerations, so we illustrate in Figure 4-30 (with much larger fonts and boxes) the additional classes and associations.
- To illustrate trigger associations, we include the PurchaseOrder commitment from the acquisitions cycle.

- A traditional bill of materials is illustrated with the SupplyItem-IssueType-RawMaterial constellation.
- Because the commitment constellation of ScheduledOperation-ProductionOrder-ScheduledIssue is inserted all together, there is only one promise association needed.

As was the case with our AAE revenue exchange example shown in Figure 4-22, readers are reminded that this M1 UML example is much simplified. A real manufacturing process would have considerably more detail.

On a last note with regard to the AAE example, readers should consult Figure 4-31 which illustrates a simple stark fact with regard to value chain modeling in a dependent-demand situation (such as just-in-time): resources at the accountability layer in REA flow forward in the value chain while commitments for resources at the scheduling layer flow backward. The output of the acquisition cycle (raw materials) is the input to the conversion cycle, but the commitment for raw material consumption (the scheduled issue) “triggers” the commitment for raw material acquisition (the purchase order).

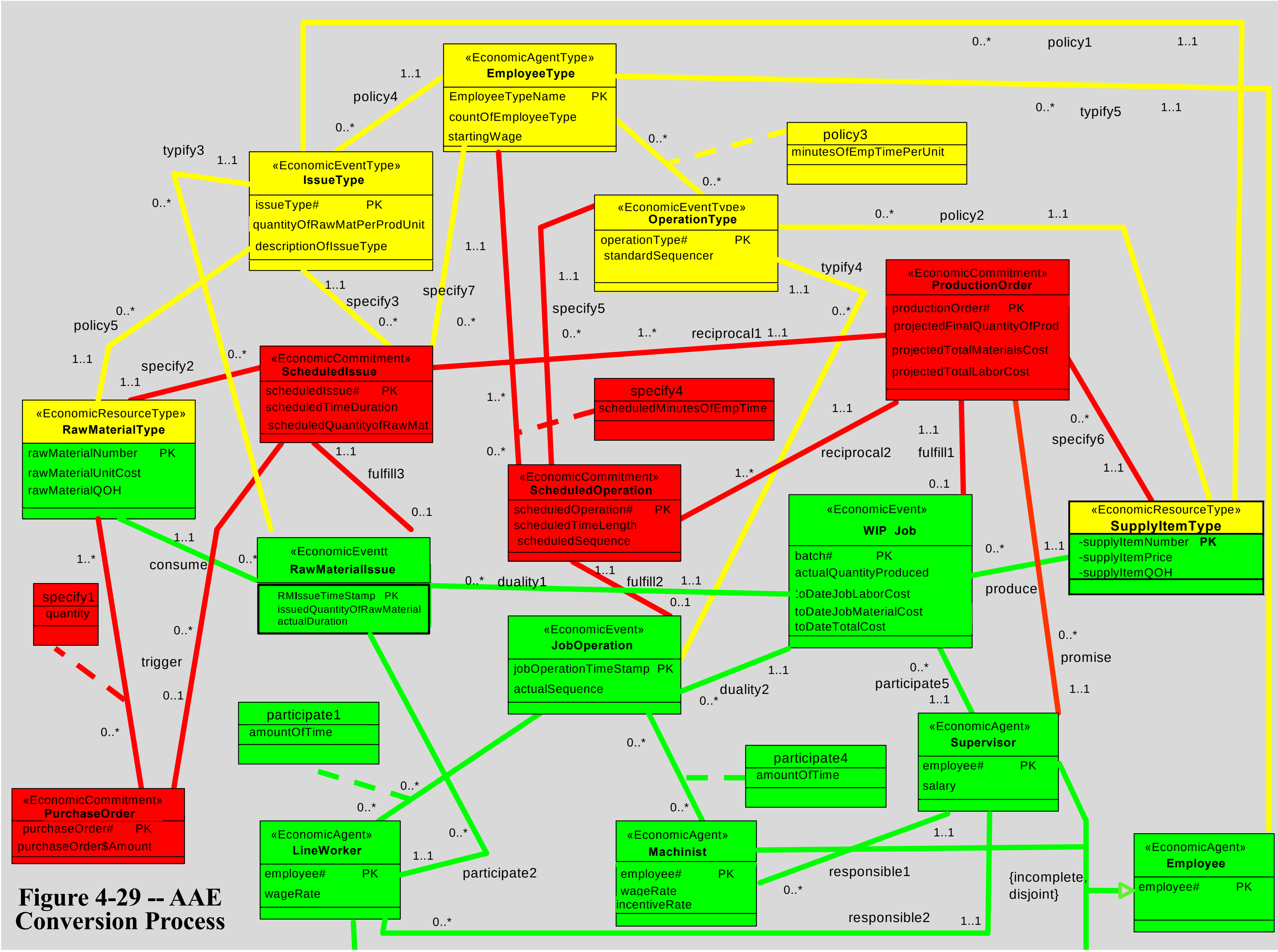


Figure 4-29 -- AAE Conversion Process

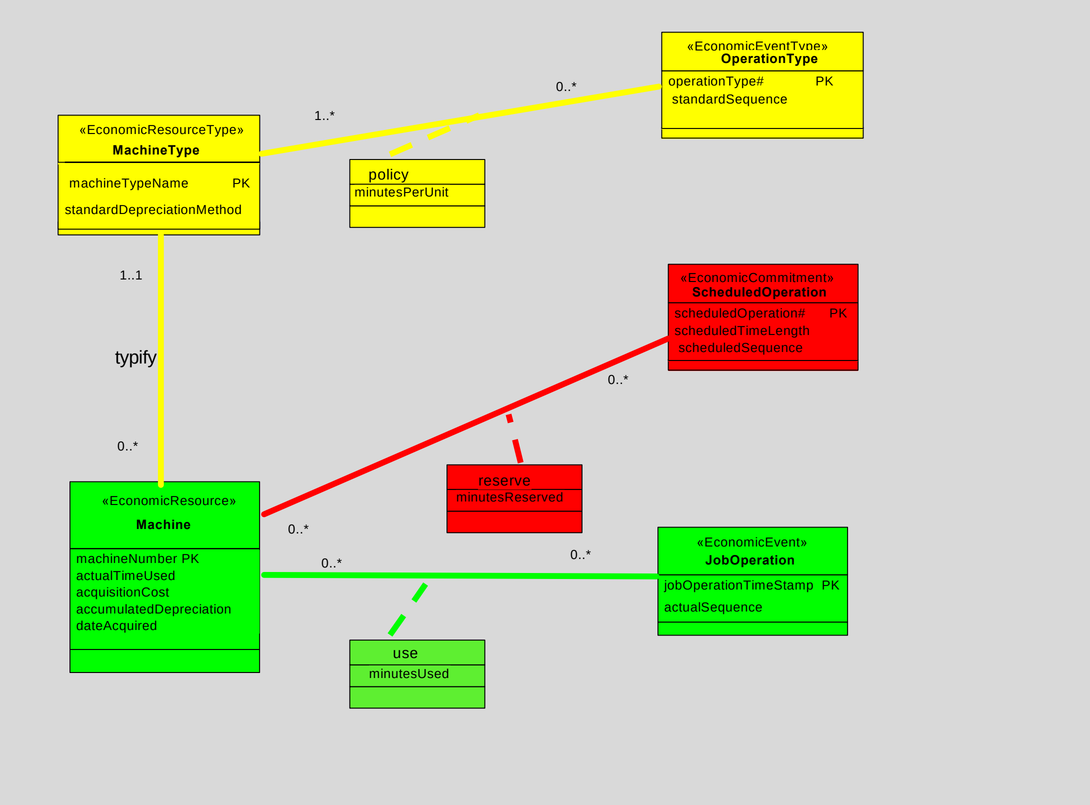


Figure 4-30 – AAE Conversion Process (continued)

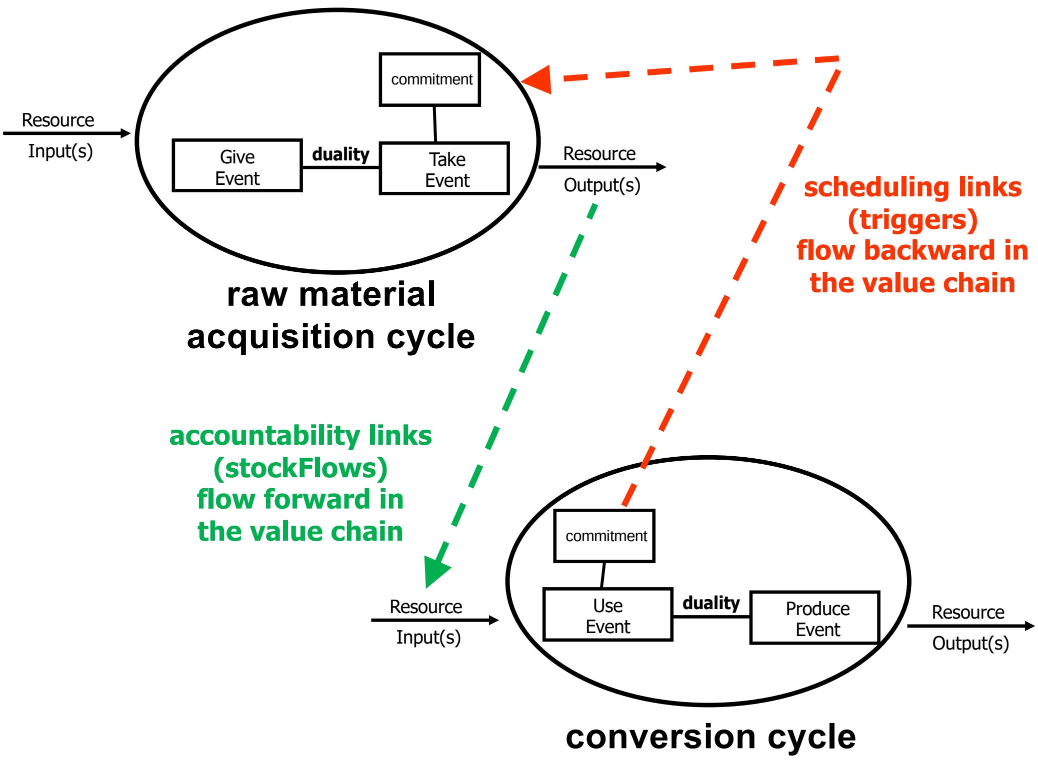


Figure 4-31 – Resource versus Commitment Flows

Conversion Patterns

The basic accountability pattern shown in Figure 4-24 can actually be applied to a wide variety of "making" activities. We can usefully think of five major prototypes for conversion modeling, but there are more to be designed as the study of REA conversion continues. Looking at these five ***patterns*** should help readers understand the essential nature of REA conversion modeling.

- The Finished Product pattern (see Figure 4-32) – The output here is normally a finished good, such as a manufacturing run of cars or a batch of cookies. However, useful management artifacts such as designs or schedules also fit. A value conclusion may be materialized about these artifacts if such a judgement is necessary (e.g., if an outside market exists) or they can be simply seen as the aggregate of their used or consumed factors of production. The output of a design process is a resource that can be used to direct subsequent production or managerial action. For example, the design of a car model "X4J" can then be used to direct the production of the X4Js. Each completed car of this particular model embodies some of the design resource.

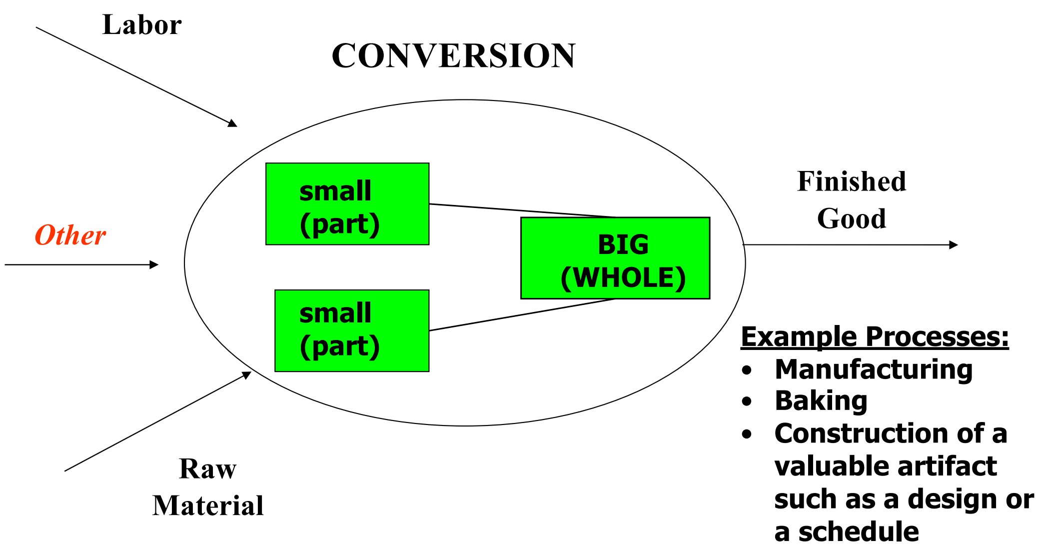


Figure 4-32 – REA Conversion Pattern #1 – Finished Product

- The Enduring Occurrent pattern (see Figure 4-33) – These occurrents use or consume many resources to fulfill the agreed upon contract. The completed large-grained "produce" process actually becomes a resource to be exchanged or used/consumed downstream in the value chain, hence the orange dotted circle which signifies conceptual congruency. If the output is separately representable, we often refer to it as a ***deliverable.*** Examples include consulting engagements, advertising campaigns, and other project-oriented jobs.

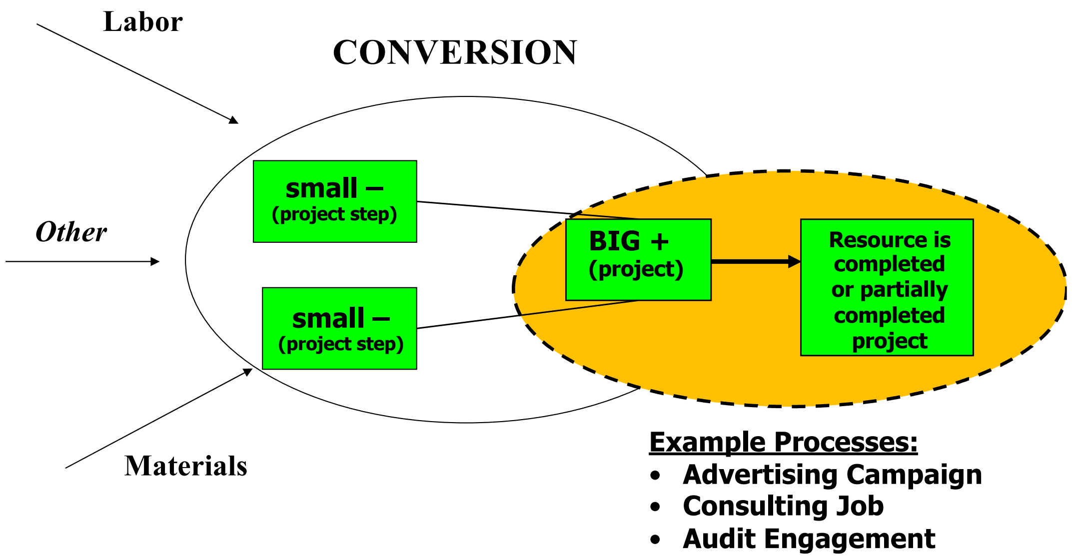


Figure 4-33 – REA Conversion Pattern #2 – Enduring Occurrent

- The Carrier Resource pattern (see Figure 4-34) – For this type of conversion, the resource may have values for certain attributes enhanced or new values assigned. Examples might include:
  - for maintenance on an airplane, its basic structure may not be changed, but its value for "engine maintenance quality" may be readjusted;
  - for pilot training, aviators may participate in an advanced certification program which allows them to fly a certain type of aircraft, and
  - for logistics, a sold product may now have a changed value for its "delivered" attribute.

The economic rationale (i.e., why do this?) for the carrier conversion is that the downstream production functions that use augmented resources now become more efficient, or they meet regulatory requirements.[34]

[34] Accountants either expense these costs directly or capitalize them depending on how the carrier resource is enhanced

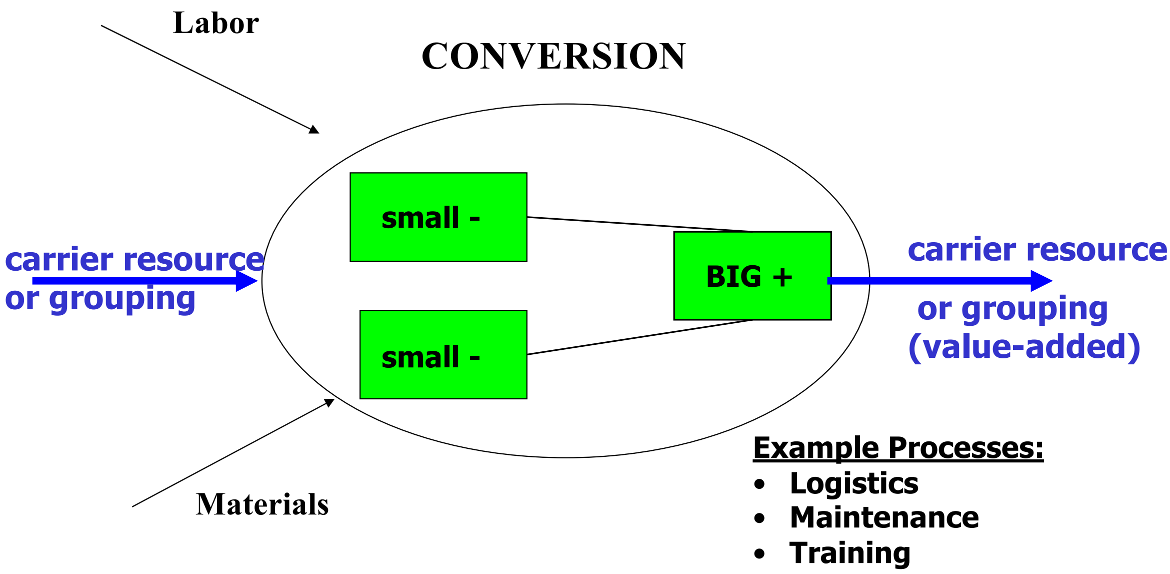


Figure 4-34 – REA Conversion Pattern #3 – Carrier Resource or Group

- The Increment/Decrement Congruency pattern (see Figure 4-35) – For this type of ***one-step*** conversion, the decrement, and increment events are conceptually congruent. Examples of this include the cleaning of fish where the fish resource is input and converted to cleaned fish, or the inspection of a car where the output is an inspected car. In each case the resource's value is increased by a single labor event.

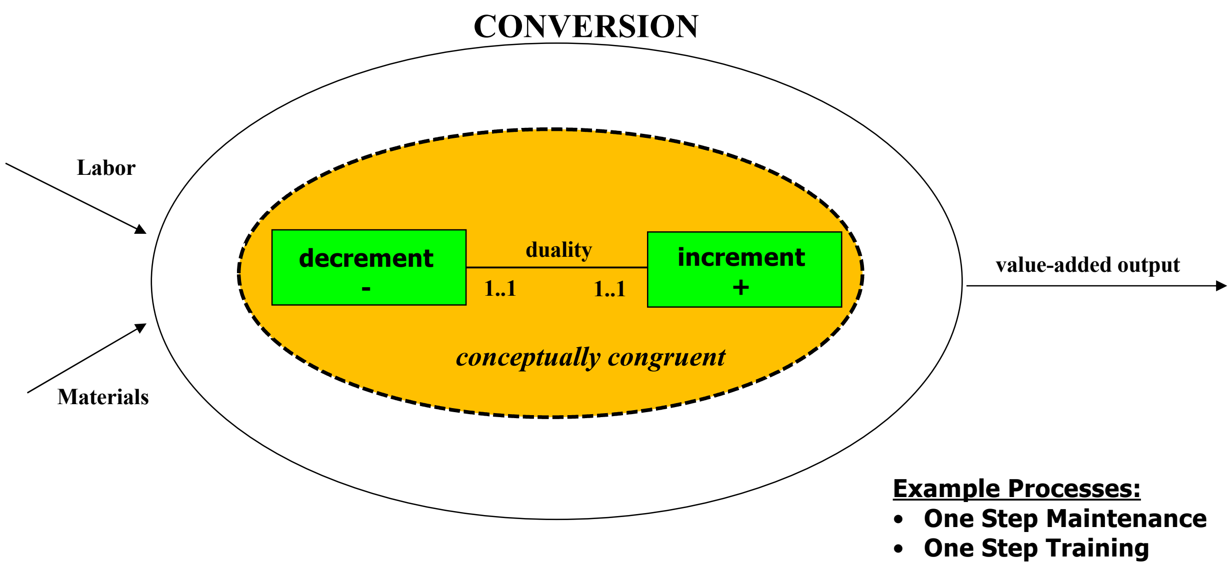


Figure 4-35 – REA Conversion Pattern #4 – Decrement-Increment Congruency

- The Participation Rights Partitioning pattern (see Figure 4-36) – This prototype can be thought of as an inverse of normal conversion where the whole occurrent is broken down into individual participation rights. So, a company may produce a music concert and then provide individuals with the right to attend and consume the music at the concert. The production of the concert consumes various resources such as the venue, the advertising, the band's labor, and the security for the event. The concert resource is then partitioned into tickets or rights to consume the resource. This pattern is a good example of how such artifacts may arise, as it shows a different path to the AAE revenue process representation. Patterns arise when modelers see the solution of a somewhat unique business process as a template for other processes. In this chapter, the derived AAE solution could serve as a basis for modelling ship cruises and music concerts.

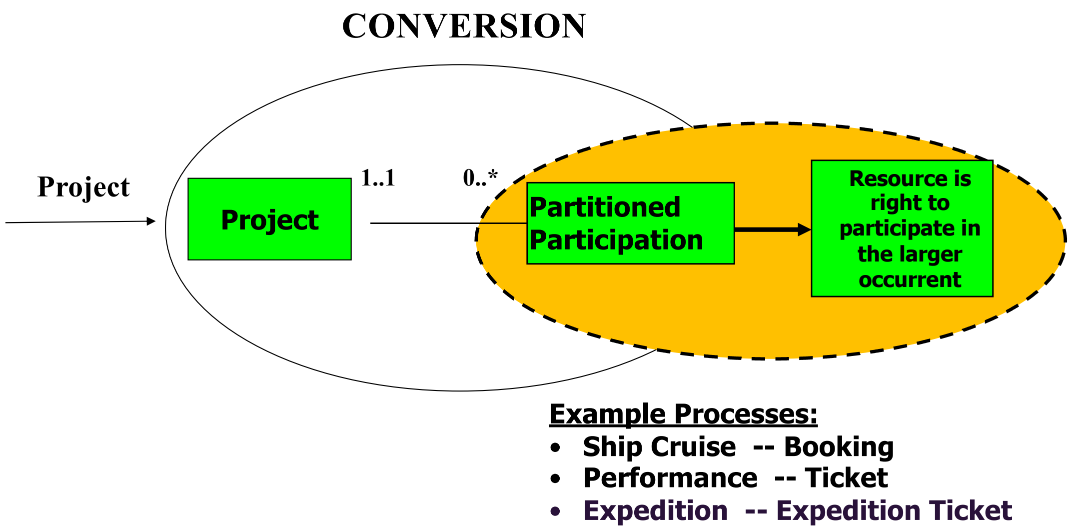


Figure 4-36 – REA Generic Conversion Pattern #5 –Participation Rights Partitioning

As mentioned above, we expect more prototypes to materialize in the future as REA conversion is studied more extensively. Indeed, the entire concept of developing design patterns for such prototypes is an area of intense study in the object-oriented community (Hruby, 2007; Hruby & Kiehn, 2006). Some initial design pattern work for REA exchanges was started but not finished by Geerts and McCarthy (1997b), and it needs to be developed further for both exchanges and conversions.

Chapter Summary

In previous chapters, normative and informative components of the REA ontology were presented in some detail. The initial focus of this chapter however, was on possible implementation issues where contractions and expansions to certain REA components are considered. The primary reason for contractions is an inability to find identifiers for economic phenomena, and the term "conceptual congruency" was presented as a basis for contracting multiple classes into a single representation. For expansions, the term "meronym" was used to describe the division of certain classes into whole-part structures. Following these initial implementation considerations, the rest of the chapter considered issues and structures involved in defining the differences between REA exchanges and REA conversions. Since the extended description of conversions was a new path in the REA literature, it was given extra emphasis, and we continued with a new set of conversion patterns for readers to contemplate and possibly use. We finish this chapter with a reiteration of some key concepts.

- ***REA market exchange*** -- a business process in an external market where two parties with independent economic interests engage in matched transfers of economic resources (like a purchase of raw materials for a disbursement of cash).

- ***REA internal conversion*** – a business process internal to an enterprise where employed economic agents transform a set of input resources (like raw materials) into a set of output resources (like finished goods).
- ***Business Process phases*** – the start-to-finish groupings of ***business events*** that mark the progress of a ***business process.*** Defined phases for a market exchange are planning, identification, negotiation, actualization, and post-actualization, while those for an internal conversion are planning, negotiation, actualization, and post-actualization.
- ***give*** and ***take*** – these are subsets of REA ***stockFlow*** associations used for exchanges. A ***give*** relationship indicates that an enterprise or entrepreneur transfers an economic resource to an outside party, while a ***take*** relationship indicates the dual reverse flow of an economic resource into an enterprise from an outside party.
- ***use***, ***consume***, and ***produce*** -- these are subsets of REA ***stockFlow*** associations used for conversions. Inflows to a conversion process are either ***consumed*** (like an egg for a cake) or ***used*** (like an oven for a cake). A ***produce*** relationship connects the large-grained economic event in a transformation to the resulting economic resource (a final product or deliverable).

Chapter 5: The Independent View for the REA Ontology

The Evolution of Accounting Systems toward Shared Inter-Enterprise Data

Dunn and McCarthy (1992) outlined a vision for past, present, and future accounting systems in a categorization scheme that emphasized the differences between industrial age accounting systems and information age systems. Eleven years later, David, McCarthy, and Sommer (2003) extended that vision to the enterprise as a whole in a sweeping review of the enterprise software systems then presently available (and soon to be available) in the marketplace. The focus on this chapter is to demonstrate the ability of the REA ontology to represent business processes and business events in this setting; something which is not possible with the traditional accounting model. In Figure 5-1, we combine those two perspectives, so we can discuss the effect that both the REA ***trading partner view*** and the REA ***independent view*** will have on new accounting software implementations. Our discussion will focus especially on those whose components which will reside outside the boundaries of a typical enterprise in the setting known as *Collaboration Space* (International Standards Organization, 2007).

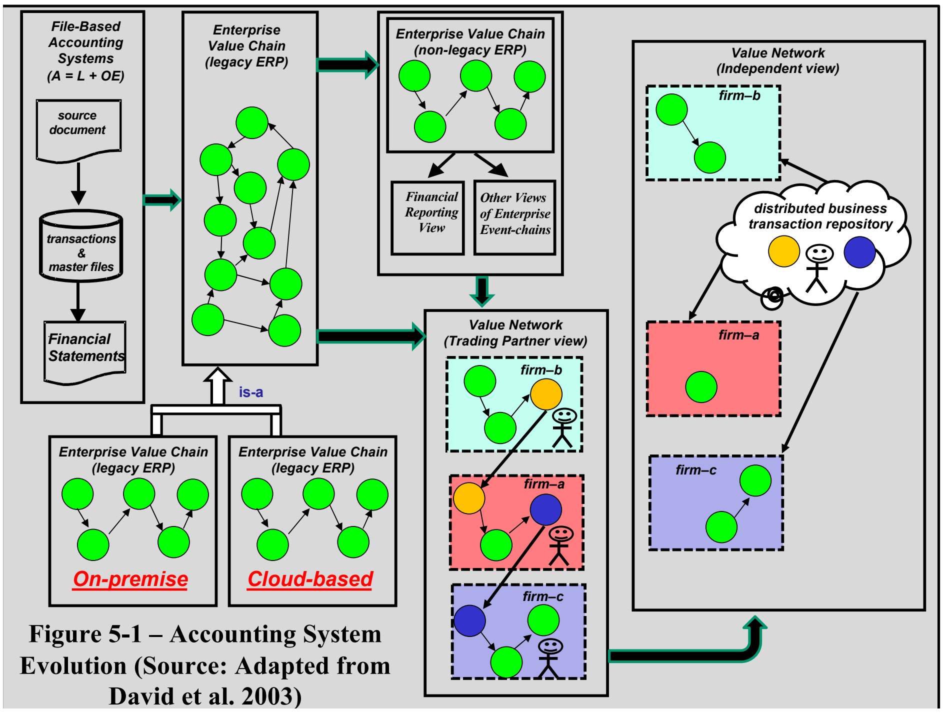


Figure 5-1 – Accounting System Evolution (Source: Adapted from David et al. 2003)

From left to right in Figure 5-1, readers may discern these five categories of accounting systems as adapted from Dunn and McCarthy (1992) and David, McCarthy, and Sommer (2003), and as augmented by a current assessment of conceptual advancement possibilities.

1. **File-Based Accounting Systems (ALOE) systems** – The fundamental accounting equation (**A**ssets = **L**iabilities + **O**wners **E**quity (ALOE)) was the organizing principle for paper-based double-entry bookkeeping systems that existed before computerized data processing began to be used heavily in the 1960s. This certainly influenced the file-based accounting systems of the first 20 years of the computer age. Those software systems imitated special and general journals with transaction files (centered

on Sales, Purchases, Cash Receipts, Cash Disbursements etc.), and they imitated ledgers with master files (centered on Accounts, Employees, Customers, Vendors, Finished Goods, Raw Materials, etc.). They mimicked posting by perpetual or periodic updates to an extensive General Ledger file. These systems predominated in the 1970s and 1980s, and they are still common, albeit on a much smaller scale.

2. **Enterprise Value Chain (Legacy ERP) systems** – These kinds of enterprise-wide information systems are based (in theory at least) on an integrated company value chain that links business processes like Purchasing and Payroll all the way through to Sales Order Entry systems, including the trail of value-creation activities through the manufacturing or conversion process. We label this category as “legacy ERP” because the integration is/was achieved by connecting (from the bottom up) older limited-purpose components from accounting, operations, marketing, and logistics, instead of by beginning anew (from the top down) with a fresh value chain analysis template. For accounting purposes, these legacy platforms retain many double-entry artifacts as mainstream features, including bloated charts of accounts with long faceted account codes plus absorption, periodic, and activity-based costing schemes. These all drive causal entrepreneurial logic which must be passed through a general ledger before its materialized conclusions can be used for decision making. Use of these legacy ERP systems exploded in the 1990s, and they still currently dominate the market (in modestly modified forms). Their legacy essence was seen by many to be modified when cloud-based ERPs (also known commonly as software-as-a-service ERP) began to emerge in the early 2000s. That however was a mistaken conclusion as both subclasses of ERP systems (on-premise and cloud-based) generally chose to maintain their legacy infrastructures.

3. **Enterprise Value Chain (non-legacy ERP) systems** – As their name implies, these kind of value-chain-based systems were designed top-down with a mind toward limiting the effect of older, limited-purpose artifacts on the overall architecture by relegating the artifact’s materialization and use to a database view (as opposed to being part of the core database set of elements). For example, (1) linking advertising use to actual sales can be analyzed through the perceived causal links with marketing logic, instead of through a periodic scheme that assumes that expenses in a certain period produce revenues in that same period, or (2) the relationship of two inputs to a production function can be analyzed by examining the behavior of the actual phenomena instead of correlating one input’s ledger account balance behavior to the other’s. These are the kinds of systems organized around the REA principles first espoused in McCarthy (1982), and examples of actual implementations at both the prototype level (Gal & McCarthy, 1986) and the commercial level (Nittler, 2018) have shown repeatedly that financial reporting can be accomplished quite well by a minimal general ledger with other double-entry artifacts limited to database views (instead of letting those artifacts in at the base transaction level).

4. **Value Networks (Trading Partner View)** – In their software evolution analysis, David et al. (2003) reserved their highest categorization for what they termed community systems or networks of trading partners who try to communicate with upstream vendors and downstream customers in a somewhat standardized way with both predictable formatting of inter-organizational transaction data and a limited ability for one trading partner to peer into the enterprise system components of another (and vice-versa). For software, this often meant bolting Advanced Planning

and Scheduling (APS) and Customer Relationship Management (CRM) modules on top of the ERP software. In Figure 5-1, we show these relationships by connecting the acquisition process of the middle enterprise (firm-a) with the revenue process of its upstream vendor (firm-b) and by connecting the revenue process of the middle enterprise with the acquisition process of its downstream customer (firm-c). As shown by the arrows between the category boxes, these value networks can evolve from both legacy ERP architectures and non-legacy ERP architectures, and it is our contention that the linkages will proceed smoother for the REA-type systems because the double-entry artifacts of the trading partners will not have to be synchronized or dealt with in some other ad hoc manner. David et al. (2003) agree here by stating that it is much more likely that trading partners will choose detailed business semantics (the REA foundational principles) rather than bookkeeping structures for transfer protocols. It is also extremely important for readers to note here that market exchange information in trading partner value networks is most commonly modeled and stored twice, once by each party who obviously have different relative perspectives on the data (as depicted by the different stick figures in each company).

5. **Value Network (Independent View)** – In the first version of the Open-edi Economic and Accounting Ontology, ISO 15944-4[35] (2007), the concept of "collaboration space"[36] was introduced as a place where transactions take place and are modeled above and separate from a community of trading partners. This space is reserved for modeling and recording market exchanges that affect multiple trading partners, and most importantly, the data is modeled only **once** even though it ultimately triggers updates in the systems of multiple enterprises. This is illustrated in the far right box of Figure 5-1 where two exchange processes have been moved into collaboration space and where the middle enterprise has both its acquisition process and its revenue process lifted up to the common area where processes are viewed from an **independent** perspective. Also, Figure 5-2 illustrates informally how this movement occurs, and it highlights the need for established procedures that will materialize two trading partner views from the one independent view. As readers may see, enterprise neutral terms like "transfer" can be mapped to a "purchase" in one firm's ERP system and to a "sale" in another firm's ERP system.

[35] ISO 15944-4 is based on the REA ontology.
[36] The term collaboration space was introduced into ISO 15944-4 by contributions in 2001 and 2002 by the Japanese delegation to Open-edi led by Katsuhiro Morita (ISO, 2007).

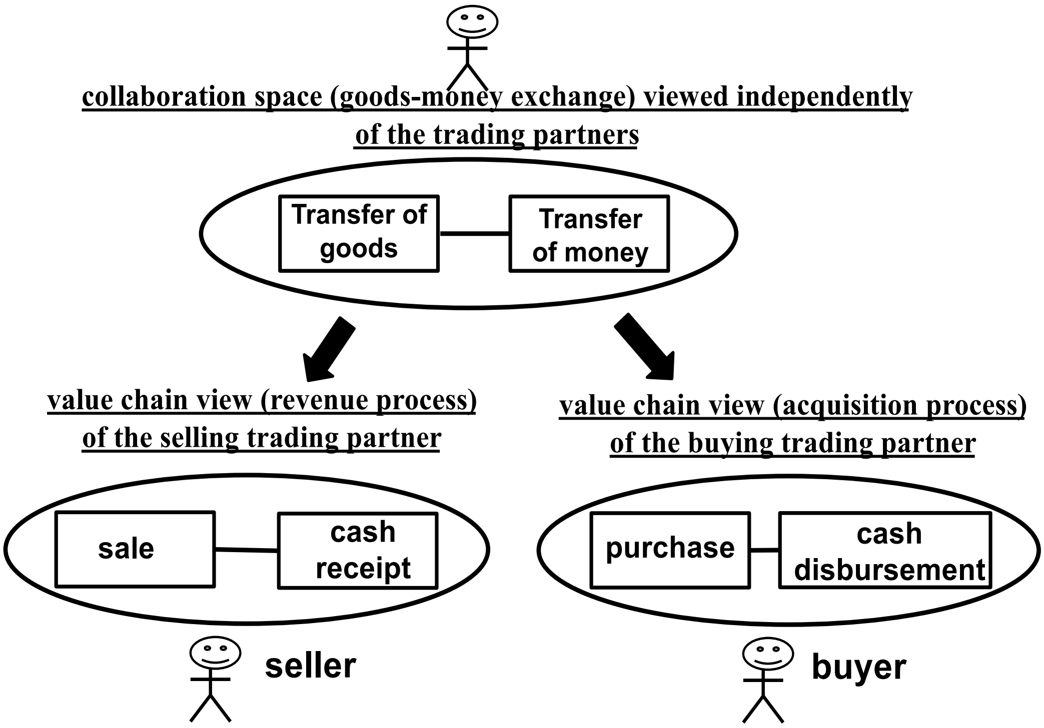


Figure 5-2 – Different Views of a Goods-Money Exchange

The last of the accounting/enterprise systems illustrated in Figure 5-1 (Value Network – Independent view) are the subject of this chapter. Traditional double-entry systems are extremely limited (if not completely infeasible) in this space because the foundational accounting equation (Assets = Liabilities + Owners Equity) is inherently the trading partner view of just one enterprise: one partner's sale is another's purchase as shown in Figure 5-2. For the REA ontology however (as indicated above), this reorientation is relatively straightforward. Indeed ISO 15944-4 (2007) was written from this very perspective.

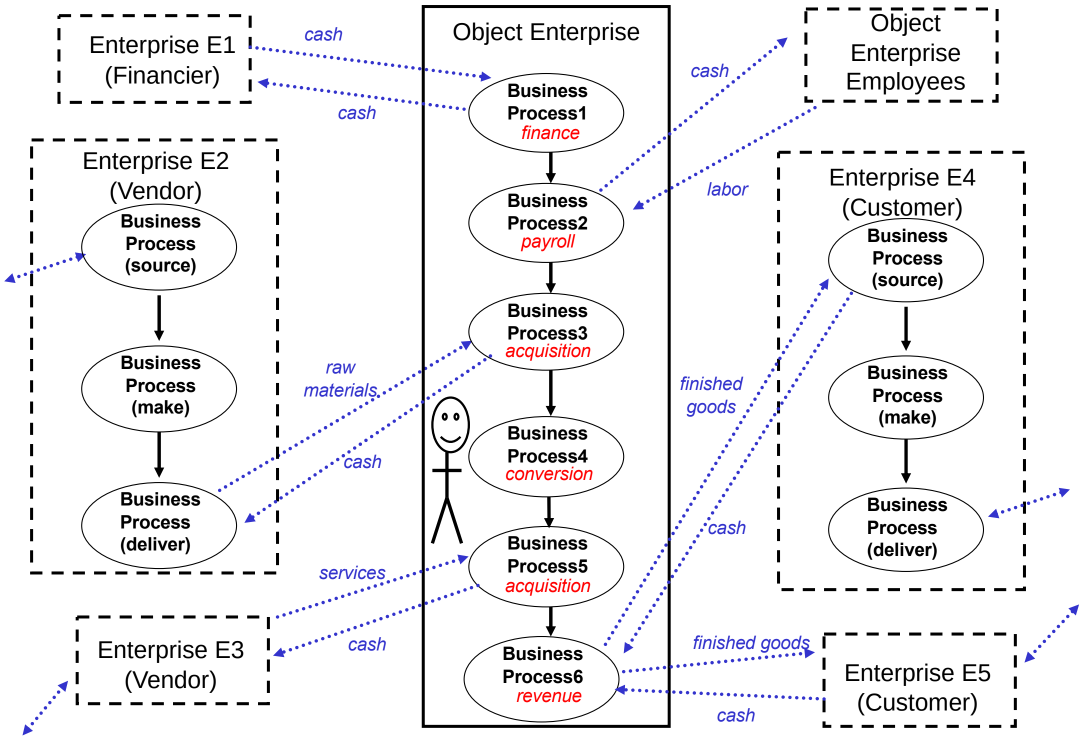
The original REA paper (McCarthy 1982) and the first four chapters of this monograph were written from the trading partner perspective however, so our task in the next few sections is well-defined: to illustrate in narrative terms how the transformation from trading partner view to independent view is accomplished.[37] To accomplish this, we will use some figures from chapters 1, 2, and 3 to review the trading partner perspective, and then we will show how these figures are reoriented to fit the independent view.[38]

A Review of the Trading Partner Perspective

Figure 5-3 is adapted from Figure 2-5 of Chapter 2. In the middle we can see that all economic phenomena that occur visibly in the owner/entrepreneur's trading community are seen from a perspective inside the object enterprise (which is delineated by a solid line). The object entrepreneurs can see all of their own business processes, and they view all inter-company exchanges from that perspective. Dotted lines connecting business partners in the value network represent flows of resources. Thus, goods bought from suppliers are viewed

[37] Algorithmic procedures for effecting the transformations from the independent view back to the trading partner view have been developed by Laurier, Kiehn, and Polovina (2018).
[38] The need for REA modeling from the independent perspective was first discussed by McCarthy (2000).

as purchases of raw materials, and goods shipped to customers are viewed as sales of finished goods.



Chapter 5-3 – REA Modeling at the Value Network Level (Trading Partner View)

Figure 5-4 is adapted from Figure 1-12 of Chapter 1, and it illustrates in an informal way (that relies on case grammars from linguistics) how the action verbs that describe economic events were derived in the original REA model from a trading partner perspective. The entrepreneur (or the agent playing the inside participate role) simply becomes the subject of the descriptive sentence, and the active verb that results describes the economic event. Thus, employee Alice purchases eggs and employee Carol disburses cash, rather than vendor Jones Company selling eggs and receiving cash. That second view for Jones would actually be tracked in the Jones Company ERP system.

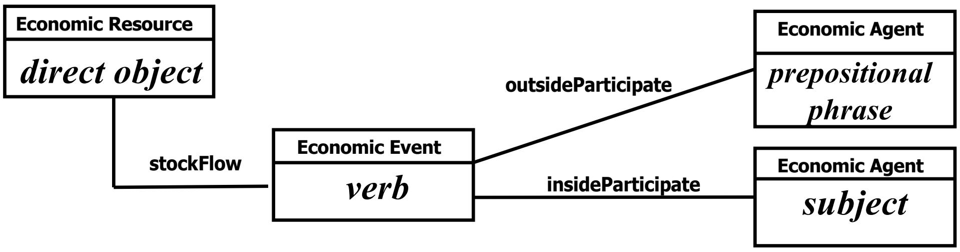


Figure 5-4 (a) – Sentence Translation of the REA Metamodel (Trading Partner View)

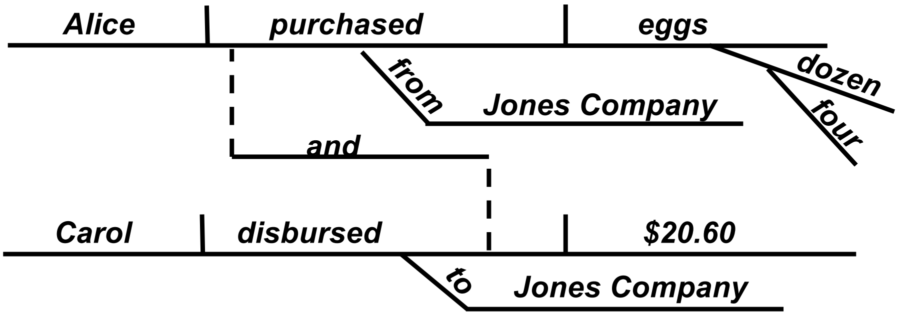


Figure 5-4 (b) – Transactions #7 and #8 Diagrammed as an REA-Patterned Compound Sentence (Trading Partner View)

Figure 5-5 is adapted from Figure 3-16 of Chapter 3. At the accountability layer, the inside/outside agent dichotomy is reflected in the *participate* associations – a direct result of such a designation (with minor modifications) in the original REA paper (McCarthy 1982). At the scheduling layer, this inside/outside distinction propagates to two places: (1) the *promise* associations between Economic Agents and Economic Commitments and (2) the *negotiate* associations between Economic Agents and Economic Contracts. In actuality, it will only be these three sets of paired associations that must change in the REA metamodel when the relative focus switches to the view of someone external to all of the trading partners. In ontological terms, this is a limited and easily accomplished switch.

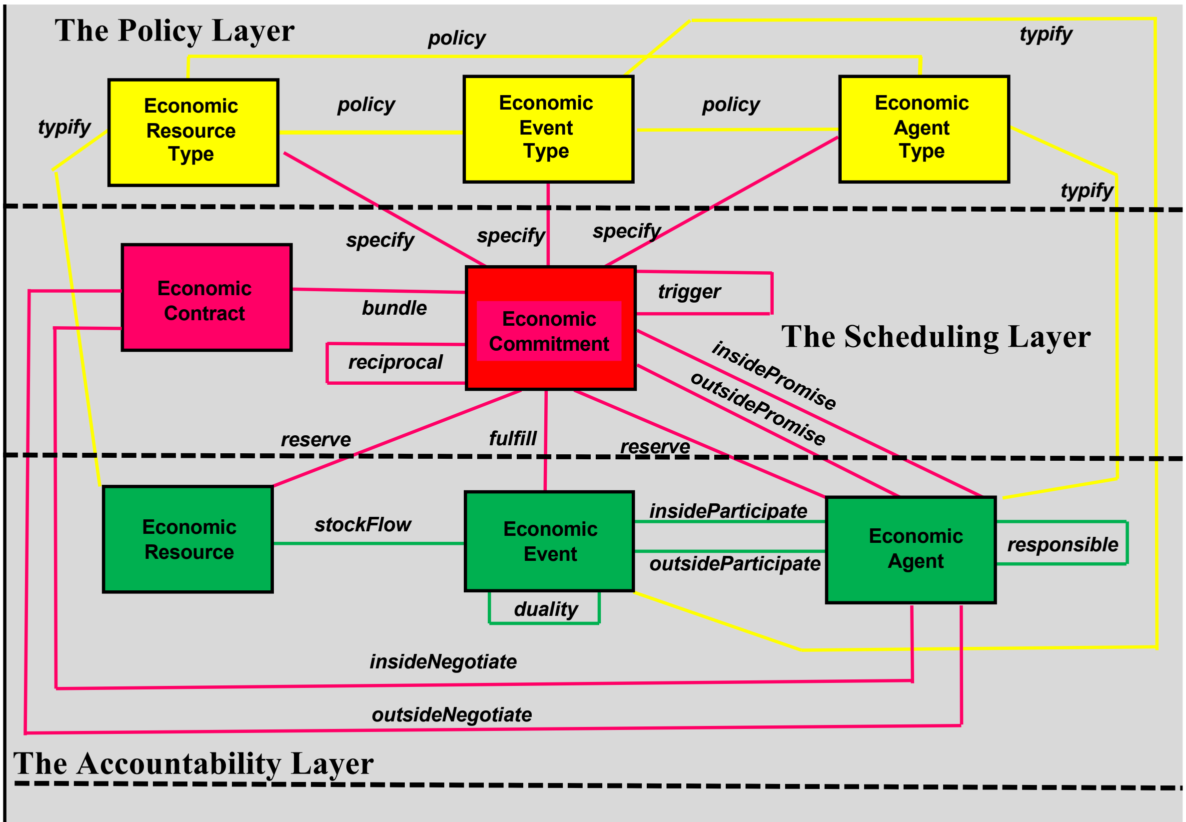


Figure 5-5 – The REA Metamodel (Trading Partner View at MOF M2)

The Reorientation to the Independent Perspective

Figure 5-6 illustrates how we designate an REA value network from an independent view, which we illustrate by moving the stick person icon outside of any enterprise boundaries. The former object enterprise around which definitions were forged now simply becomes an undifferentiated member of a larger trading community. We illustrate this by delineating it with dotted lines, and by renaming any resource flows as necessary. For example: *finished goods* and *raw materials* simply become the more general terms *goods.*[39]

[39] Again, raw material and finished goods are terms used by individuals within a specific firm; i.e. one firm's raw materials are another's finished goods.

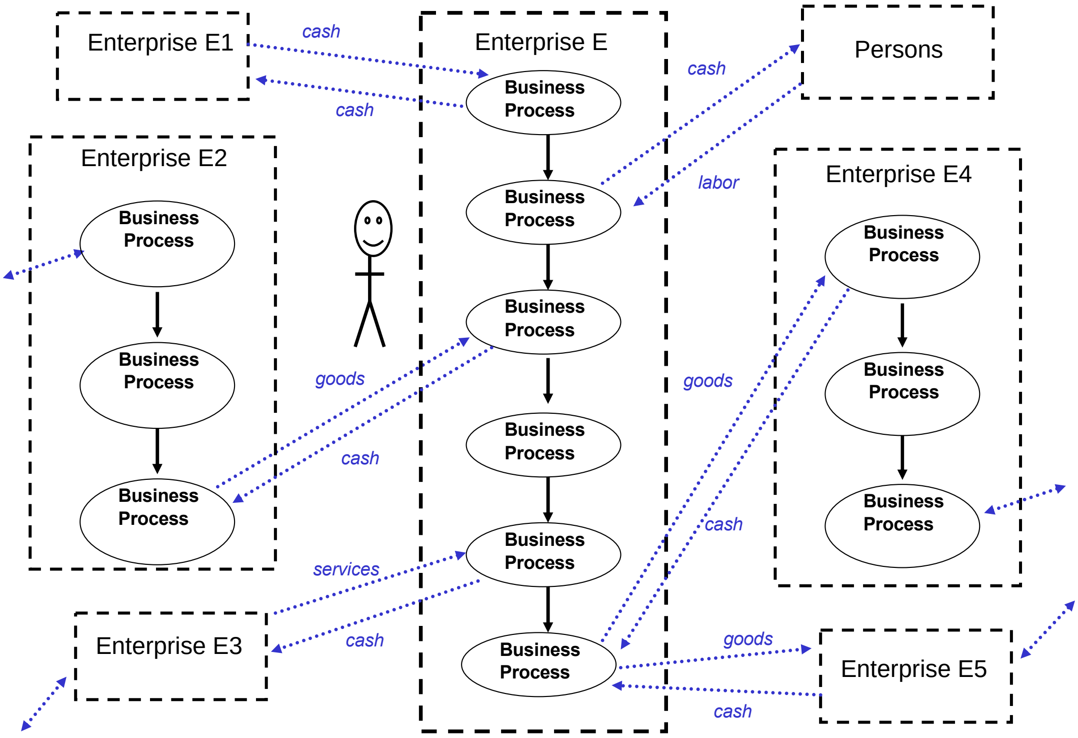


Figure 5-6 – REA Modeling at the Value Network Level (Independent View)

In Figure 5-7, we again illustrate an informal linguistic device (diagramming a sentence) to show how the relative orientation of terms has changed. The Economic Resource now becomes the sentence subject of a passive verb with two participial phrases (a *from* phrase and a *to* phrase) showing the direction of the economic transfer. This *from-to* device (as opposed to *inside-outside*) was being used informally in REA modeling for a variety of purposes before the first version of ISO 15944-4 was finally published in 2007, but as we will see in the next figure, that standard formalized this distinction.

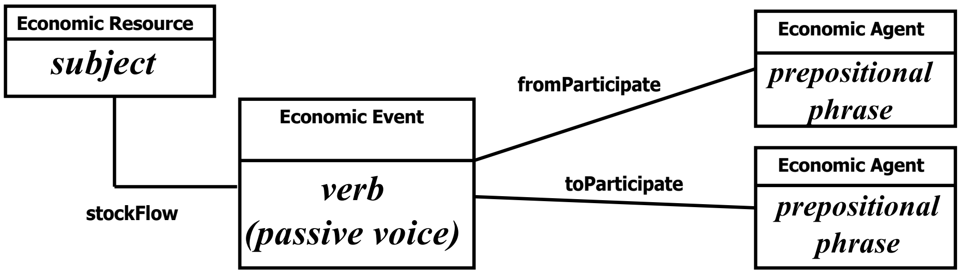


Figure 5-7(a) – Sentence Translation of the REA Metamodel (Independent View)

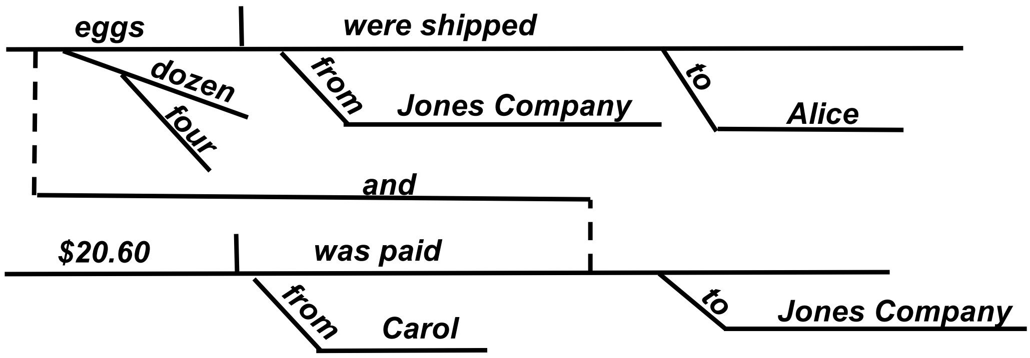


Figure 5-7(b) – Transactions #7 and #8 Diagrammed as an REA-Patterned Compound Sentence (Independent View)

Figure 5-8 illustrates the full independent view REA metamodel. Readers should note these differences.

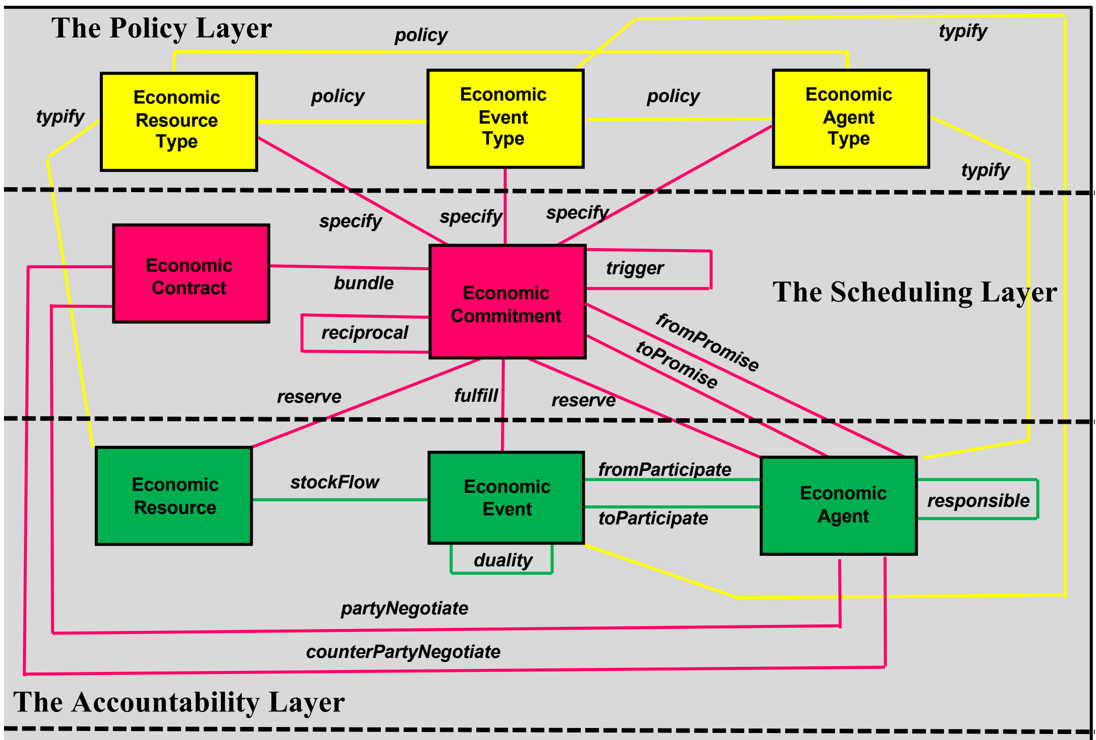


Figure 5-8 – The REA Metamodel (Independent View at MOF M2)

- At the accountability layer, ***fromParticipate*** and ***toParticipate*** have replaced ***insideParticipate*** and ***outsideParticipate***. *From* indicates the agent who is relinquishing an economic resource, while *to* indicates the agent who is gaining an economic resource. The *from* and *to* refer to two sides of the same economic event.
- At the scheduling layer, there is also a dual switch with ***fromPromise*** and ***toPromise*** replacing the former inside/outside pair. *FromPromise* indicates the agent who promises to make an individual future event actually occur, while *toPromise* indicates the agent to whom that individual commitment is made. Thus, a schedule to exchange resources indicates a promise of resources going "from" one party "to" another.
- Additionally at the scheduling layer, the *inside/outside* pairing for the negotiation on the entire contract has been changed to ***partyNegotiate*** and ***counterpartyNegotiate*****.** The actual designation of which separate company agent assumes the *party* role vs. which assumes the *counterparty* role is not important just as long as the established designation is known and maintained consistently. *Party* and *counterparty* are simply legal terms to designate economic agents with competing and independent economic interests. This describes perfectly what is occurring in collaboration space.
- Finally, ***stockFlow*** associations may no longer be decomposed below the top level of Figure 1-11 for the REA independent view, because terms like outflow and inflow are tied to the perspective of a single trading partner.

The Figure 5-8 independent view metamodel represents what we offer now as the most **general** case of the REA ontology. The trading partner view (which in fact is how the original model was designated in 1982) then becomes a **special** case to be derived in circumstances where both views are working in conjunction with each other. The inside-the-firm trading partner view that we have presented in the first four chapters will still dominate in practice, because firms will still have established boundaries whose performance metrics (including traditional financial statements) will remain important and whose business processes will almost assuredly include a large set of internal conversion processes with information components those firms consider proprietary. In that sense, the REA trading partner view becomes the conceptual foundation for the non-legacy ERP systems developing now in current practice. However, as we will discuss in the rest of this chapter, the real innovations to come in enterprise systems will feature the independent view of REA, because increasingly, more independent economic agents will be using *collaboration space* to store their market exchange information.

Distributed Business Transaction Repositories

In ISO 15944-1 (2001), a *business transaction* standard was developed that defined that term as a set of activities aimed at accomplishing a defined business goal; this definition equates strongly to what we have defined in Chapter 2 as a *business process*:

> A business process (BP) is an occurrence in time that accepts resource inputs, uses those resources in directed activities, and produces resource outputs of value to a potential customer (Hammer & Champy, 1993, p. 53).

ISO 15944-4 (2007) leveraged that definition to specify an "Open-edi Business Transaction Ontology" (OeBTO) – based on the REA model – that would govern business transactions in collaboration space. That 2007 standard was developed before the technological revolution of cryptocurrencies was introduced by Satoshi Nakamoto (2008), but the blockchain technology ideas initiated there (and developed extensively since) have created the perfect technology platform for the ideas of collaboration space and the independent view of the REA model. Worldwide and completely open (i.e., non-permissioned[40]) blockchains are still in the infancy of their development (Tapscott & Tapscott, 2016), but we will examine here the potential operation of the REA ontology in ***distributed business transaction repositories*** (International Standards Organizations, 2019). Figure 5-9 illustrates how a distributed business transaction repository could work with the ontological concepts we have already discussed. The business processes in each firm's value chain would be divided into two categories:

1. Private processes to include all conversions plus those market exchanges not transacted in the distributed repository. These would be governed by the REA trading partner view, perhaps as implemented in an ERP system.
2. Public processes to include exchanges transacted in collaboration space. Those business transactions would only be recorded on the blockchain with an REA independent view of the ontological categories, but their effect would be aligned with the individual firm trading partner view via transformation rules of the type pioneered by Laurier et al. (2017).

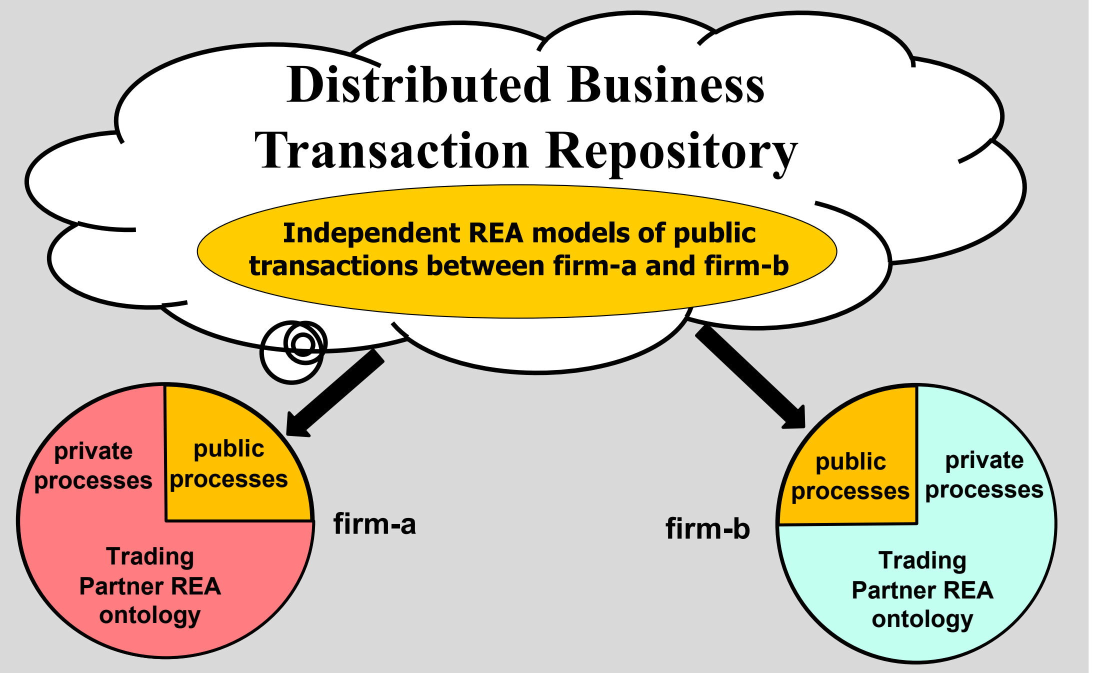


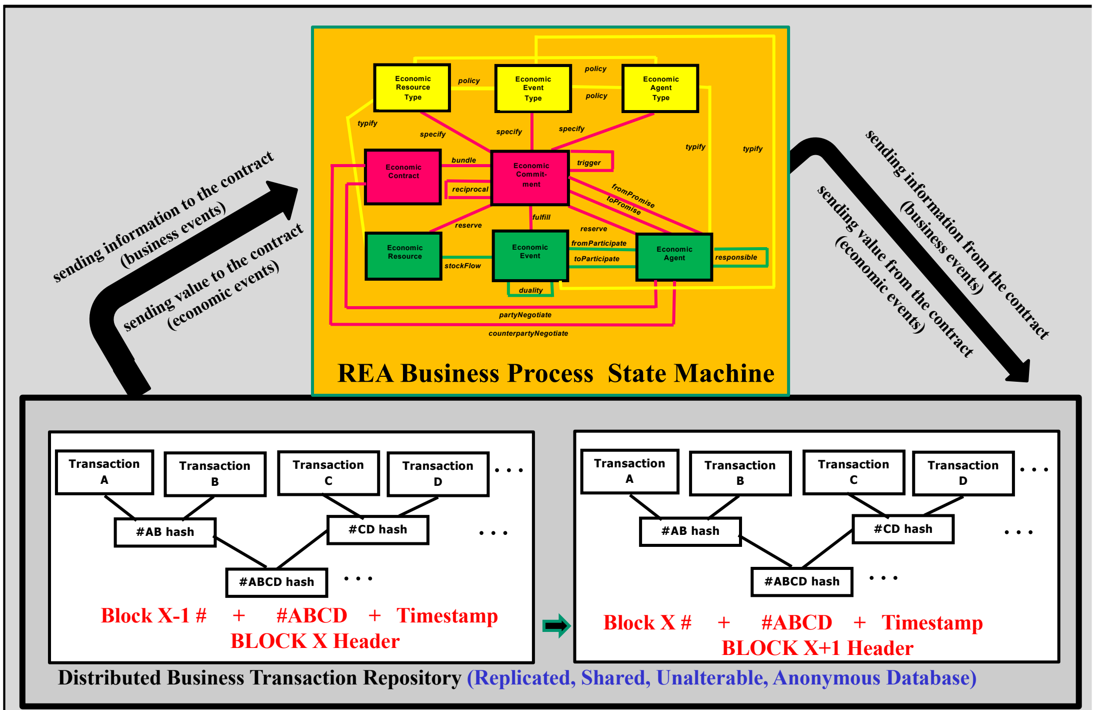
Figure 5-9 – Collaborative Storage with a Distributed Business Transaction Repository

The individual firms could use their integrated trading partner view to produce materialized conclusions like financial statements, but their exposed exchange data would not be maintained under their control. It would instead be on the distributed business transaction repository.

Figure 5-10 illustrates more specifically how the distributed business transaction repository works. A non-permissioned blockchain is a replicated, shared, unalterable, and

[40] A non-permissioned blockchain is open to all parties on an anonymous basis, and its methods for achieving consensus on valid new blocks of data are also accomplished by processes open worldwide. A permissioned blockchain is a special case where membership is limited to specific trading communities, anonymity is often foregone, and consensus mechanisms are only open to limited members of the trading community.

anonymous database whose blocks of data are stored in a long, unbroken chain where each block is verified and proved at very short time intervals. The transacting agents are anonymous parties, and their negotiations and dealings with each other are governed by blockchain-contracts which are simple instances of independent view REA contracts where the counterparty (the economic agent usually acting second in an exchange) foregoes custody of the to-be-exchanged resource to a software agent on the blockchain. That software agent (which now has become a trusted third party to the exchange) monitors the behavior of the first party in the provision of the good, service, or right to the counterparty. Such monitoring includes accounting for REA business events which are the workflow details that emanate from the trading partners and/or recognition of other business events that originate in the data gathering devices of the internet of things. The device for monitoring the progress through the entire business transaction (i.e., being able to see what the status of the "deal" is between the party and the counterparty) is an REA business process state machine – a mechanism described initially in McCarthy W. E. (2003b) and ISO 15944-4 (2007) and explained in more detail for a particular instance in Horiuchi and McCarthy (2011) and Bergholtz, et al. (2004). As will be discussed in Chapter 6 of this monograph, REA state machines are an underdeveloped area of research, but their basic functioning can be explained briefly.



Source: Adapted from The Economist (31 October 2015) and from Richard Brown (2015)

Figure 5-10 – A Distributed Business Transaction Repository State Machine

A blockchain contract is an instance of the REA independent view metamodel where each class is represented by a business object (i.e., a category instance or a token). Those objects have a defined life-cycle of states associated with them that show their progress toward becoming complete. For example, the "Economic Resource Type" at the top left of the metamodel may go through this life cycle: candidate, planned, identified, proposed, specified, and actualized (ISO 15944-4, 2007, p. 29). Progression through the state machine life-cycle is triggered by a workflow of business events, and in the most general case, the business transaction proceeds through the process phases described in chapter 4 – planning, identification, negotiation, actualization, and post-actualization.

The REA Independent View and Open Value Networks

In Figure 5-9, only a portion of the value chain processes of an enterprise becomes exposed outside of the firm by being maintained in a public repository. A more general example of the need for independent views of ontological categories is the phenomena of *open value networks* – the case where virtually all the processes shown in Figure 5-9 are moved up into the public collaboration space and become completely open and available. This is a phenomenon also called peer-to-peer production -- a value chain variant explained by Yochai Benkler in his influential paper "Coase's Penguin, or, Linux and 'The Nature of the Firm' (2002)."

Benkler's explanations are quite detailed, but they can be presented quickly by using an extension to the work of Ronald Coase, which we explored in Figure 4-10. That extension is illustrated in Figure 5-11. The "penguin" shown on the lower left in Figure 5-11 is also the penguin of Benkler's title. That penguin is the internationally recognized symbol for the Linux computer operating system – a superbly-engineered software project created and maintained not by a traditional technology company, but by an open peer-to-peer network of programmers worldwide. The main point of Benkler's argument is that such arrangements may become much more common as products become more varied, non-standardized, and service-oriented. Figure 5-11 illustrates that point by amplifying the normal make-buy decision to include a third alternative which is opening up the participation in certain business processes to the whole world to be executed in peer-to-peer fashion instead of being organized as a value chain for a single firm. Benkler especially stresses that the scarcest input resource in value creation chains in the future will tend to be human creativity, and that it makes little sense therefore to limit the source of such creativity to a limited number of hierarchically-arranged employees. Figure 5-12 and Figure 5-13 illustrate both the similarities and the differences between a traditional company value chain and an open value network.

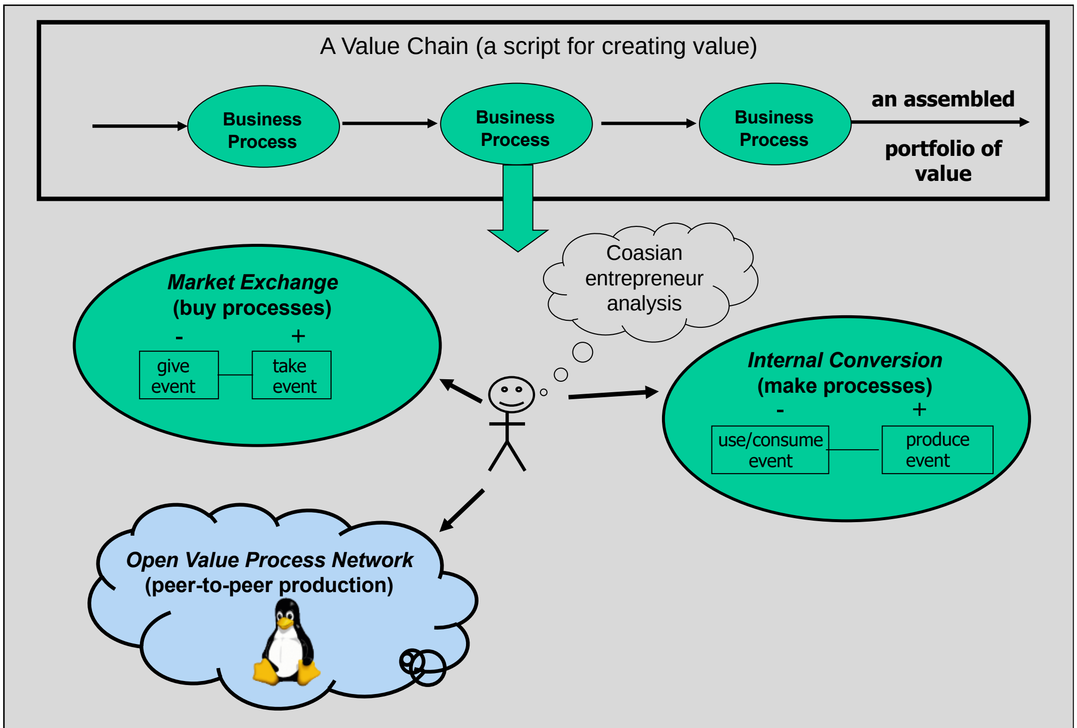


Figure 5-11 – A Third Coasian Alternative

- The value chain of Figure 5-12 starts with three market exchanges on the left; it then proceeds through two conversion processes; and it concludes with a market exchange where the assembled portfolio of value is traded for cash which in turn finances further acquisition activities. Employees and vendors are given immediate and fixed compensation for their inputs at rates determined by the entrepreneur and the marketplace. Notice that the economic phenomena are viewed from within a firm.

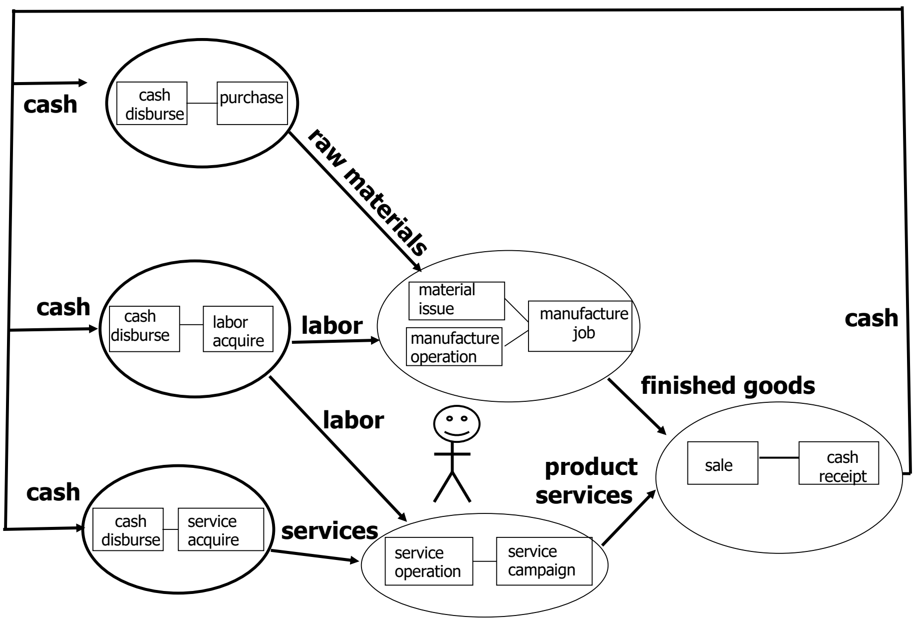


Figure 5-12 – Generic Value Chain for Goods, Services, and Labor Combination

- The open value network of Figure 5-13 starts with three nodes (a person or a small independent group of people) who contribute goods, labor, and services to two value-creating and integrating projects which in turn are integrated into a completed project of assembled final value. That final portfolio of value is then transferred to a party outside the network who in turn renders payment. That payment is then pushed back up the network and distributed to network participants in accordance with a "value equation"[41] that everybody has engineered together and agreed to. The value equation accounts for contributed economic resources like goods, labor, and services, but it also factors in things like ideas, prototypes, and designs.

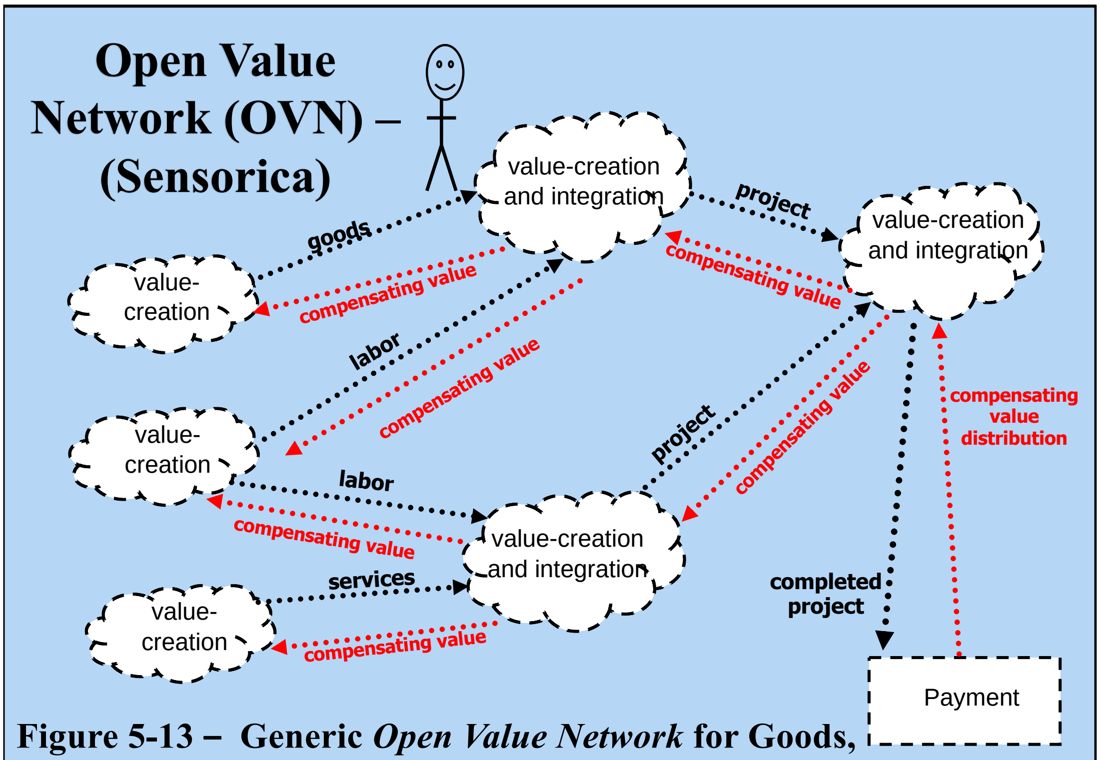


Figure 5-13 – Generic *Open Value Network* for Goods, Services, and Labor Combination

Benkler emphasizes that peer-to-peer production works best when projects have these characteristics:

1. they are highly modular,
2. they allow heterogeneous but primarily finer granularity levels of contribution, and
3. they permit low-cost, quality-controlled integration.

These characteristics are multipurpose, disaggregate, multidimensional, and highly integrative, all components of the desired-feature list, which guided the design of the original REA model by McCarthy (1982), so it is no surprise that open value network consortiums like Sensorica (2019) and Value Flows (2017) have found REA useful in accounting for both their value creation activities and their value distribution activities. In particular, it is interesting to note that the descriptions, purpose, and principles of the open-source Value

[41] One of the most prominent examples of an open value network is Sensorica – a Montreal-based consortium that pioneered the development of "value equations".

Flows consortium, as illustrated in Figure 5-14, were derived from many of the REA principles explained in this monograph's previous chapters. Readers should note especially:

- principle #4 which asks for explicit support for fractal structures of the type illustrated in Figure 2-3; and
- principle #5 which asks for modeling at the recipe, planning, and accounting levels. These ideas are based explicitly on the policy, scheduling, and accountability layers of the REA metamodel of Figure 5-8.

Value Flows Description and Purpose: *Value Flows* is a set of common vocabularies to describe flows of economic resources of all kinds within distributed economic ecosystems. Its **purpose** is to enable internetworking among many different software projects for resource planning and accounting within fractal networks of people and groups.

Value Flows Principles:

1. the vocabulary must support coordinating work between different people in different organizations using different software on different platforms using different human and programming languages.
2. the vocabulary must track the flows of resources (value) forwards and backwards.
3. the vocabulary must distribute income (rewards) according to peoples' contributions, regardless of where and when in the network those contributions occurred.
4. the vocabulary must be fractal: it must support high-level views of networks as well as zooming into lower and lower levels of detail.
5. the vocabulary must work at the recipe, planning, and accounting levels.
6. the vocabulary must work for alternative and traditional organizing shapes and economic relationships

Figure 5-14 – Value Flows Principles
(Source: ValueFlows Project 2017)

Open value networks have two major assemblages of business processes. As illustrated in the Value Flows examples of Figure 5-15 and Figure 5-16, the purpose of the first process group is to track the **creation** of value, while the purpose of the second group is to track the **distribution** of value back up the network to its participants. A general REA model for the creation of value is based on the REA conversion pattern described in Chapter 4, while the general REA model for distribution of value is based on the exchange pattern (also described in Chapter 4). Illustrations of these two assemblages are given in Figure 5-17 (the accountability layer), Figure 5-18 (the policy layer), and Figure 5-19 (the scheduling layer), and they are explained below.

Here is the resource flow for creating the product..

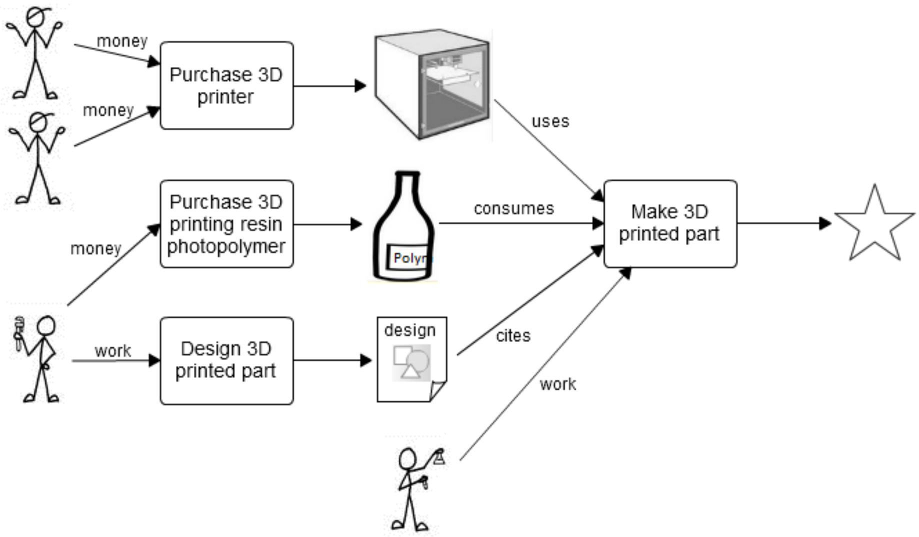


Figure 5-15 – OVN Instance of Creating Value (Source: Foster 2017)

Here is how the income gets distributed...

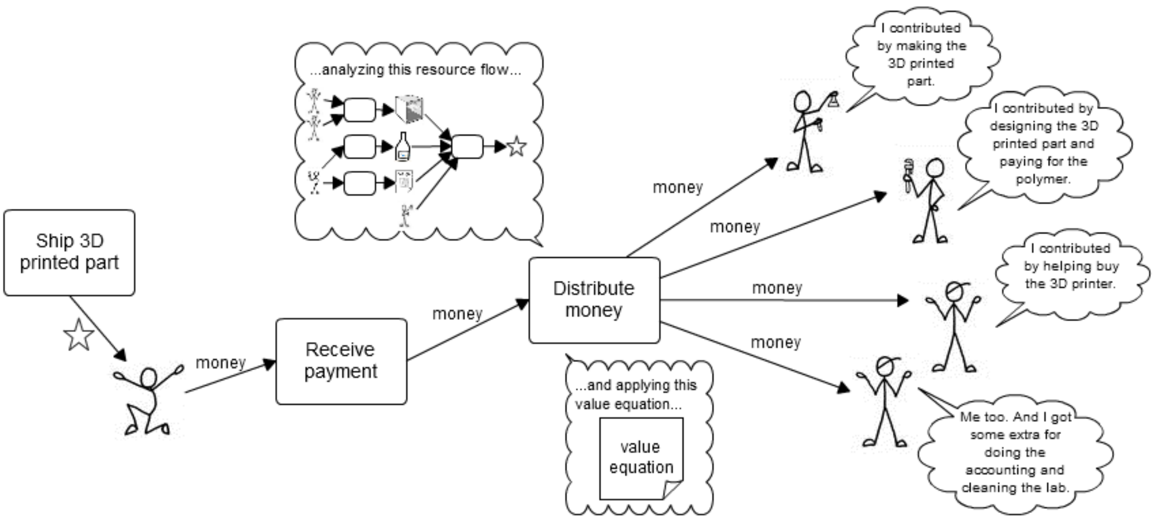


Figure 5-16 – OVN Instance of Distributing Value (Source: Foster 2017)

Figure 5-17 – What actually occurs in an example open value network (accountability layer)

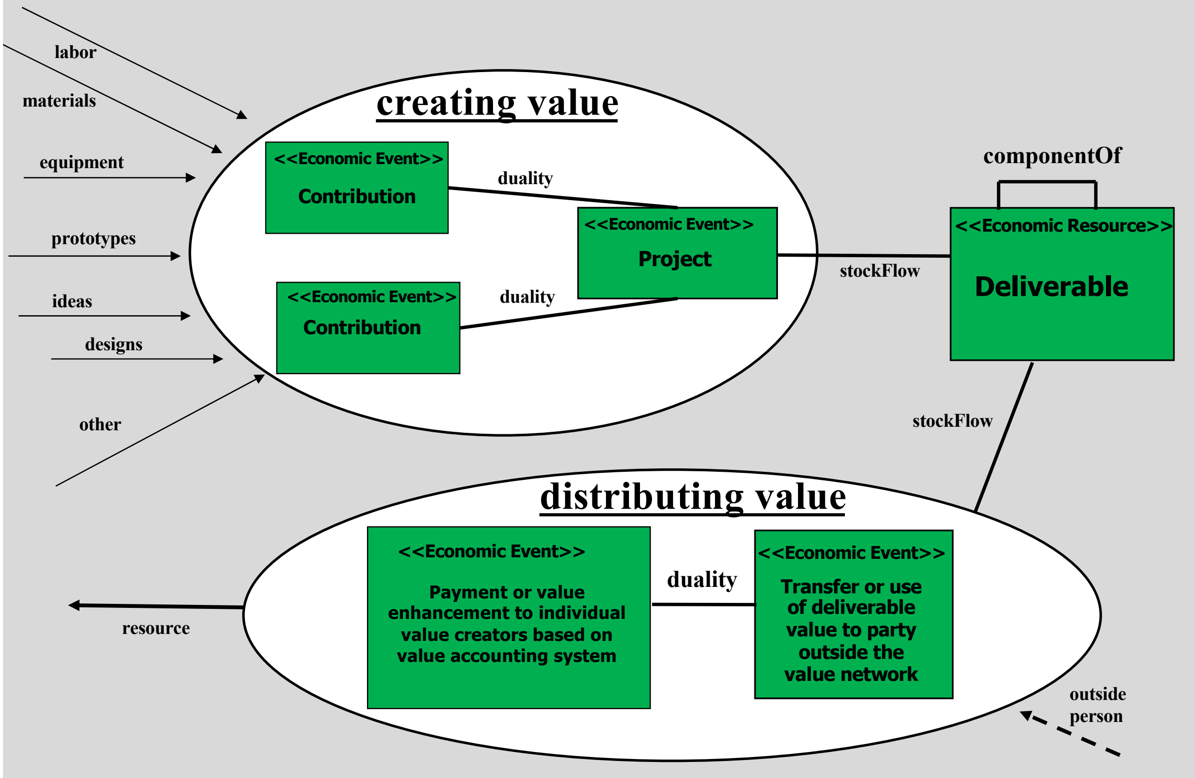
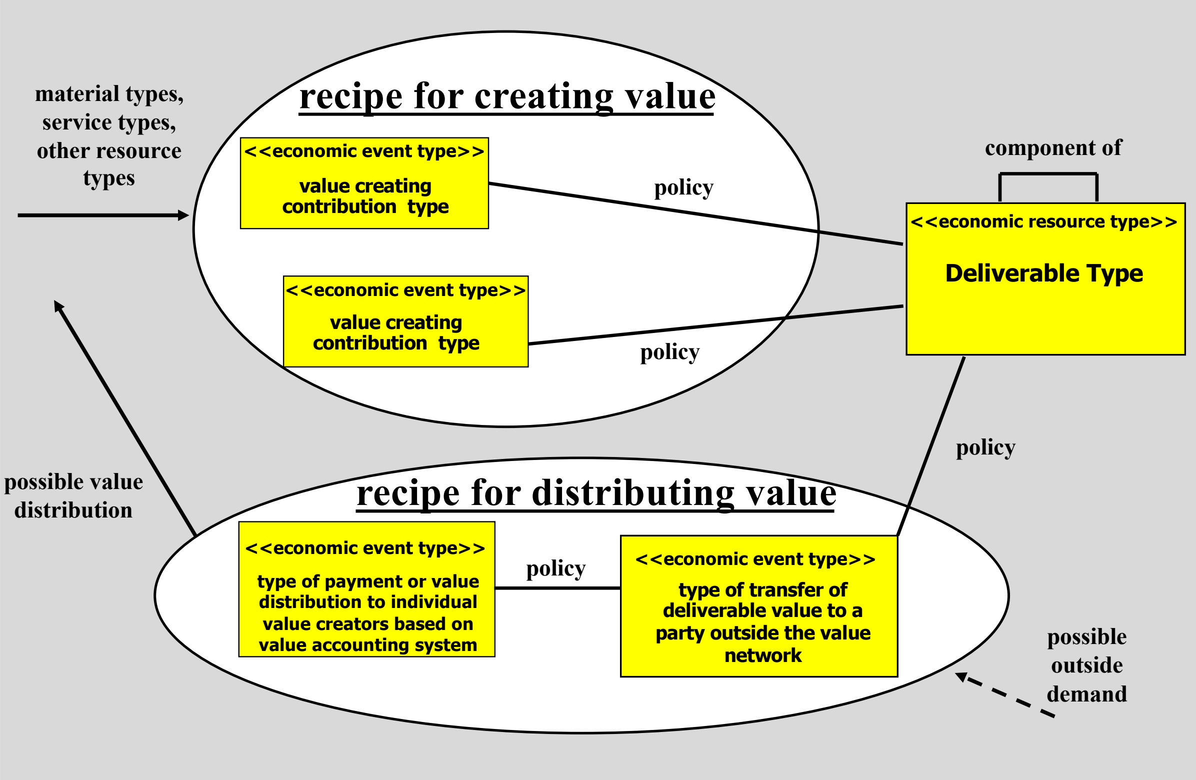


Figure 5-18 – What could occur in an example open value network (policy layer)

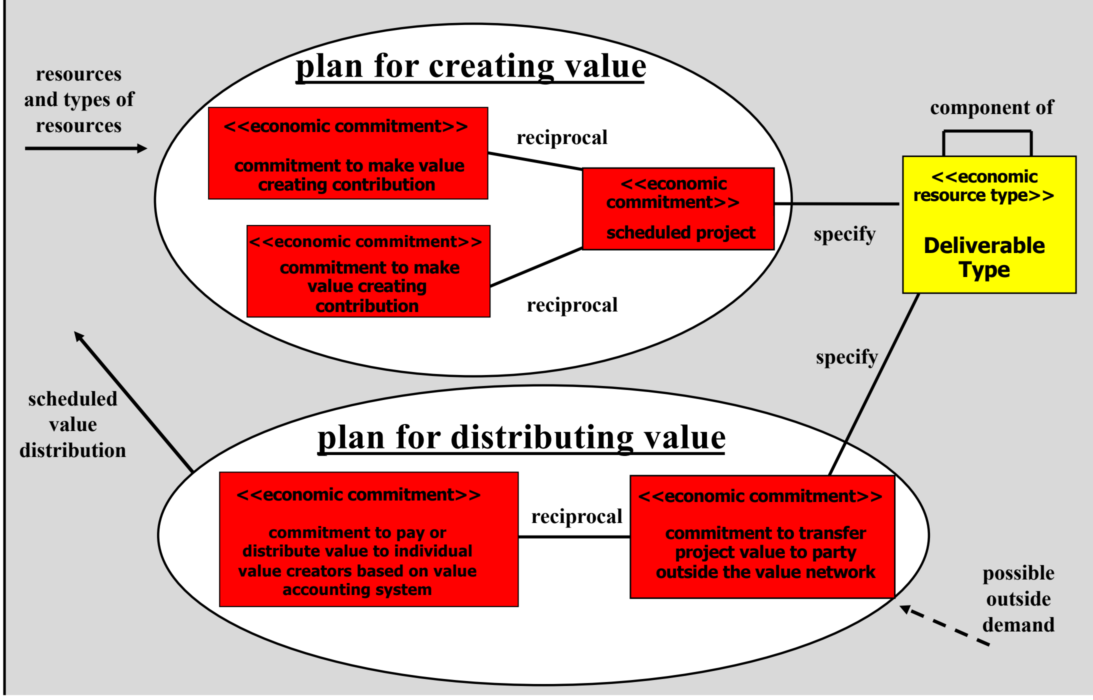


- At the accountability layer, creating value uses the generic conversion patterns of either Figure 4-32 (for goods) or Figure 4-33 (for services). The deliverable shown is a completed project which is certainly the most common case in an open value network, but it could also have been an assembled good. The contribution events on the left of the creating-value process track the project steps used in the value accounting equation to distribute value at the bottom. Readers should note that deliverable resource has a component structure: deliverables may be composed of other (smaller and more granular) deliverables.

- The policy layer of Figure 5-18 will not always be present in an open value project, but once the network gains experience with certain types of deliverables, their **recipes** for creating value can be assembled in a repository that will guide future projects. Following (Geerts and McCarthy 2006), policies can take different forms including targets (e.g. industry best practices), and validation rules (e.g. at what time a project will be classified as a “dry hole”[42]).

- The scheduling layer of Figure 5-19 demonstrates how projects are planned once the policy recipes are specified. A scheduler at the hub of the network will issue plans for value creating commitments for a particular scheduled project in an open call for participation on the network. Project completion and value distribution will follow from an open-value perspective, this becomes an even more critical view as this sets the pattern for expected or possible contributions to the network, and it is required for a system-wide understanding of the “fulfillment” process.

Figure 5-19 – What is scheduled to occur in an example open value network (scheduling layer)



[42] A dry hole is a project that does not produce a payment.

The REA Independent View – Summary

As we mentioned earlier in this chapter on the independent view of the REA ontology, we now consider this more **general** model to be the foundation of the ontology. The trading partner views espoused in most of the previous REA work (including the original 1982 paper) can now be considered **special** cases of an enterprise ontology derived from a more particular point of view. This is an extremely important view as we move from a traditional accounting model to an ontology that incorporates not only this perspective, but also models of business processes independent of any particular party. This becomes a true realization of the words in the title of the original REA paper, "A Generalized Framework for Accounting Systems in a Shared Data Environment". Below, we finish this chapter by explaining again some of its important new concepts.

- ***Collaboration Space*** -- Business activity space where an economic exchange of valued resources is viewed independently of the perspective of buyers, sellers, or other individual parties. This definition is adapted from ISO 15944-4 (2007).
- ***Independent View*** of REA – The categories, associations, and other components of the REA ontology as viewed from a universal trading-community perspective where an individual partner's view of economic phenomena is de-emphasized. Thus, the use of participant-centric M1 business and accounting terms for Economic Events (like purchase, sale, cash receipt, and cash disbursement) and Economic Resources (raw materials and finished goods) is not allowed.
- ***Trading Partner View*** of REA – The categories, associations, and other components of the REA ontology as viewed from the perspective of a particular trading partner to an exchange. This was the view of the original REA paper. All double-entry accounting models built on the "assets = liabilities + owners' equity" equation are limited to the trading partner perspective. The terms "independent view" and "trading partner view" are adapted from ISO 15944-4 (2007).
- ***fromParticipate*** and ***toParticipate*** – These are two REA metamodel associations that relate Economic Events to Economic Agents in the independent view of the REA ontology. The *fromParticipate* agent initiates the transfer of a resource to the *toParticipate* agent. The dual Economic Events flip these designations.
- ***fromPromise*** and ***toPromise*** – These are two REA metamodel associations that relate Economic Commitments to Economic Agents in the independent view of the REA ontology. The *fromPromise* agent is responsible for making the promised transfer of a resource occur to the *toPromise* agent. Analogously to the accountability sphere just above, the reciprocating Economic Commitments flip these designations in the scheduling sphere.
- ***partyNegotiate*** and ***counterPartyNegotiate*** – In the bilateral negotiation for a contract in the REA independent view, the two parties need to be designated differentially, so one of these connections between Economic Agents and Economic Contract is selected as the *partyNegotiate* association, while the other is termed the *counterpartyNegotiate* association.
- ***Distributed Business Transaction Repository*** – This is a data repository where the business and economic events involved in an economic exchange are stored. This repository may be distributed to multiple nodes in a collaboration space network, and it may also be replicated at each node. This is a definition is derived from ISO 15944-15 (2019).

Open Value Network (OVN)-- This is a Value Network where the individual nodes emphasize the open participation of individuals and small groups in peer-to-peer production and distribution activities over the participation of business firms.

Chapter 6: Extensions and Future Directions in REA

Introduction

The previous chapters discussed the fundamental components of any ontology and related them to business domains in general and to the REA ontology more specifically. An ontology for a domain must answer questions such as:

- What exists in a domain?
- What are the categories for these things?
- How are these categories related?

Our previous five chapters have defined the classes and relationships of the REA ontology – a necessary condition for any ontology (Obrst, 2003; Sugumaran & Storey, 2002). This chapter will examine the REA ontology in terms of its ability to materialize conclusions about the business domain and to create traditional accounting representations of that domain. Throughout this chapter, we present previous work that has addressed these concepts, and suggest where future research would make the current state of the REA ontology even more integrated with the business domain. We consider future research to be critical for readers of this monograph as the strength of an ontology is in its ability to extend and transform perceptions of a domain.

Our first target in this "future extensions" chapter concerns the role of business events in progressing an exchange or conversion through the phases of a business process. The mechanics of REA state machines (Horiuchi and McCarthy 2011) allow for these conclusions to be materialized. The second section covers the relationship between the declarative aspects of the REA ontology and the reasoning that can be done using these class and relationship specifications (Geerts & McCarthy, 2000a). The third section of this chapter focuses on traditional accounting disclosures and the ability of the REA ontology to directly support these disclosures. This part will also bring in work of other researchers who are using the REA ontology in their efforts to create a better description of financial instruments (Bennett, 2015; Bennett, Gilmore, & Nehmer, 2018). In section four, we discuss REA as an implementation platform for conceptual models of costing. Here, special attention is given to "the use of data abstraction (grouping) to avoid arbitrary allocation" and to "the aggregation of business events to the business process level". Following, in section five, we look at another particularly relevant accounting disclosure – the quality of internal controls. Initially, this examined automatic enforcement of internal controls was based on employee types (Gal & McCarthy, 1982b; 1985b). Later work examined the semantic basis of internal controls (Gal & McCarthy, 1991). Section six provides a brief list of further items on our engineering-oriented design agenda that we would like to see researched more extensively: standardized attributes, expansion of the base set of REA categories, and non-binary or mediated business processes. Finally, we summarize the main ideas of this monograph, and we try to portray where REA research fits with respect to other developments in the history of accounting systems.

REA State Machines

In any business there is a state of the organization that changes as events in the environment occur. For an information system whose purpose is to capture the relevant economic aspects of the organization, it is critical to have features which can capture both the information about those events as well as the classes of objects that are affected by these events. That is, there must be a connection between the states that the business goes through and the structures of the information system. Management will make decisions about which states are relevant in the organization, and therefore the events which cause these states to occur and to change. A state machine is a useful tool to represent the changes that occur as a business process progresses through its phases.

Looking back on Figure 4-8, readers may see how the top level of the hierarchy (the value network) portrays, in a cohesive fashion, how the value chains of the firms at the second level down are arranged into trading communities, and then further, how the business processes at the third level down are arranged into value chains for creating the final portfolio of attributes of customer value. Analogously, **state machines** specify how the workflow of business events (fourth level) are orchestrated into a cohesive choreography for a business process (third level) as it proceeds through its phases of planning, identification, negotiation, actualization, and post-actualization. Figure 1-11 illustrated how REA models answer the *what*, *when*, *who*, and *why* questions of business process modeling; REA state machine mechanics answer the *how* question by orchestrating the workflow of business events.

The REA state machine work had its origins in the COOL (Haugen, 2002)[43] architecture of UN/CEFACT (the United Nations Centre for Trade Facilitation and Electronic Business) and the closely allied earlier work on ebXML (Electronic Business Extensible Markup Language)[44] in the years 2000-2002. Later, in work with the UN/CEFACT group on Uniform Business Agreements and Contracts (UBAC), McCarthy (2003b) illustrated a complete state machine model for an entire exchange process. His 2003 example set of business events closely approximated the task set shown in Figure 4-19, a workflow example later adjusted and employed in ISO 15944-4 (2007). That standard defined the general state machine mechanics with illustrations adapted and explained here as Figures 6-1, 6-2, 6-3, and 6-4.

- When an application of the REA metamodel (Figure 5-8) is implemented at Level M1, each category becomes a business transaction entity[45] that has one or more life cycles of states. Transitions into and out of those states are effected by business events as shown in Figure 6-1.

---

[43] Participants on the COOL (Commitment Oriented Orchestration Layer) project for e-business included Robert Haugen, William McCarthy, JohnYunker, Jamie Clark, Jim Clark, and Christian Huemer.
[44] See Walsh (2002).
[45] Readers are reminded that an REA ***business process*** is equivalent to the ISO term “business transaction” (International Standards Organization, 2010).

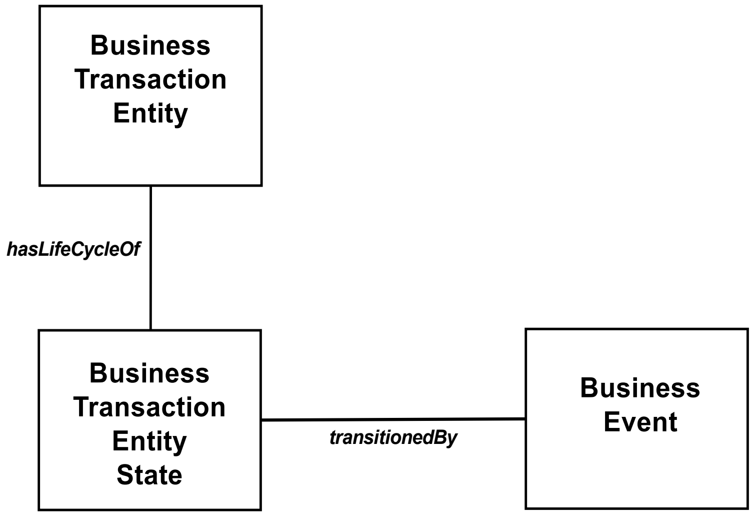


Figure 6-1 – REA Entities with states
Source: Adapted from ISO/IEC 15944-4 (2007)

- Each life cycle can be illustrated with a state machine diagram as illustrated in Figure 6-2, using UML notation, for an application of the REA ontological category ***Economic Resource Type***. This state machine has six stages, each of which is transitioned by a particular business event.

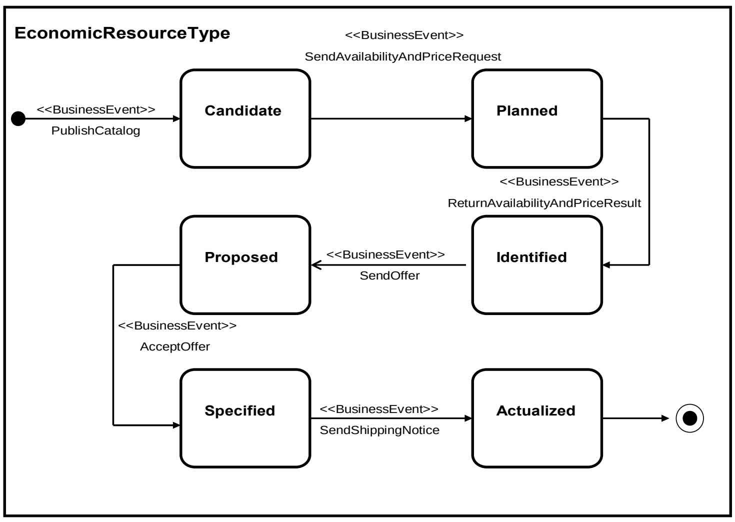


Figure 6-2 – UML State Machine Diagram for Economic Resource Type
Source: ISO/IEC 15944-4 (2007)

- The state machine mechanics are also viewed from a more macro level by activity diagrams as shown partially in Figures 6-3 and 6-4. Readers should note the correspondence between the first four events of the state machine diagram and their corresponding analog in the activity graphs.

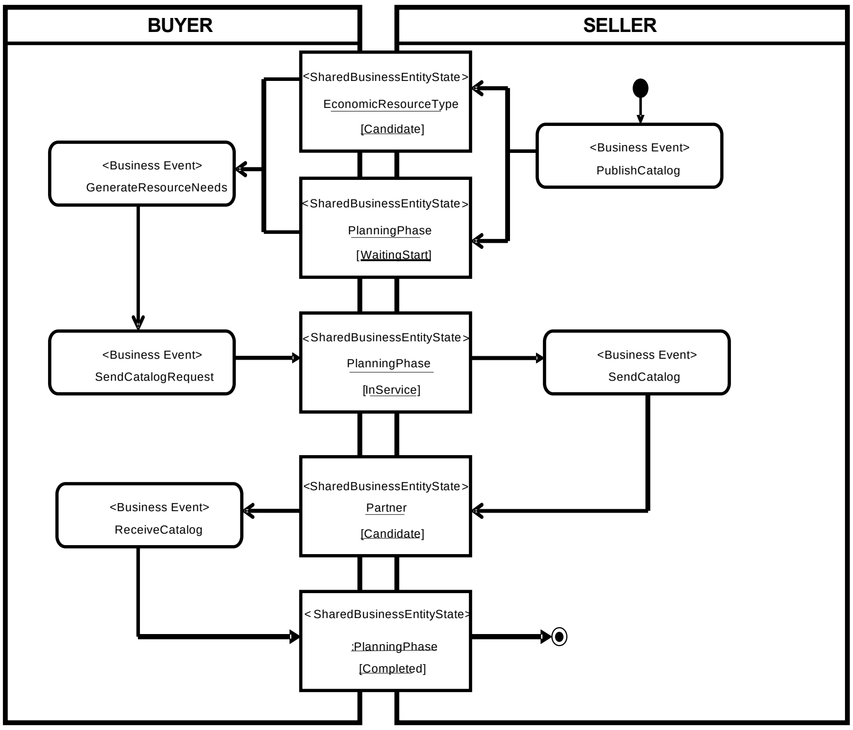


Figure 6-3 – UML Activity Graph (1) for Collaboration Source: ISO/IEC 15944-4 (2007)

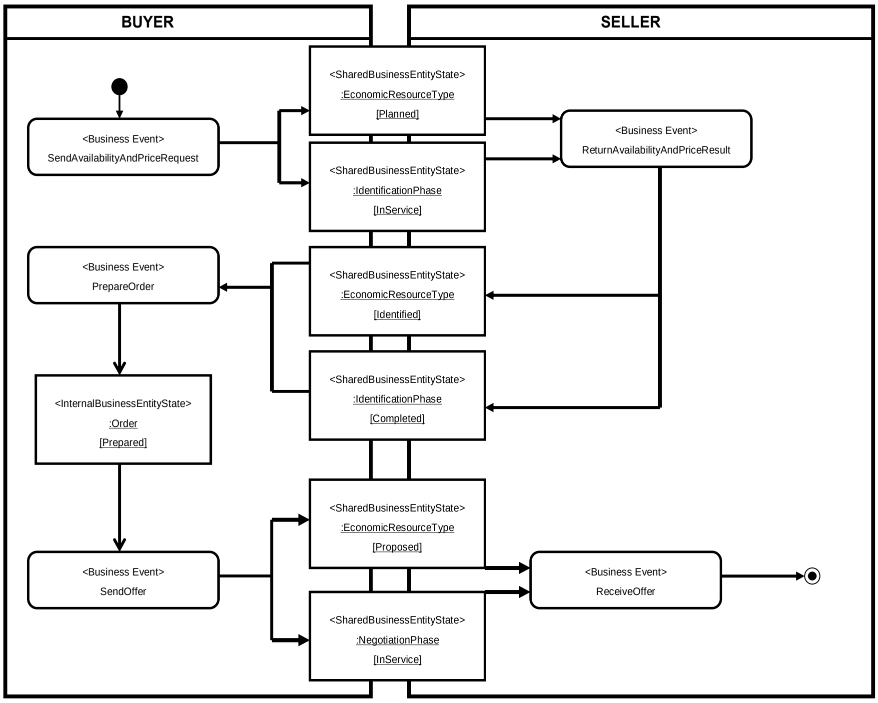


Figure 6-4 –UML Activity Graph (2) for Collaboration Source: ISO/IEC 15944-4 (2007)

A more complete illustration would need more state machine diagrams for the other business transaction entities[46] and a more complete activity graph for the entire application of the business process.

Horiuchi and McCarthy (2011) followed the ideas of ISO 15944-4 in constructing an actual state machine implementation,[47] using the object-oriented programming language JAVA.[48] Figure 6-5[49] from their paper illustrates the effect of a receiving report instance (a business object) on individual instance states of a selling agent, a buying agent, a resource type, a commitment, an economic resource, and an economic event. Figure 6-6 overviews their entire business process state machine by illustrating all of their transitioned states after the sending of a receiving report. A query or an intensional reasoner of the type to be discussed next in this chapter could use the data structures of Figure 6-6 to answer managerial questions such as: "where are we in the deal?"

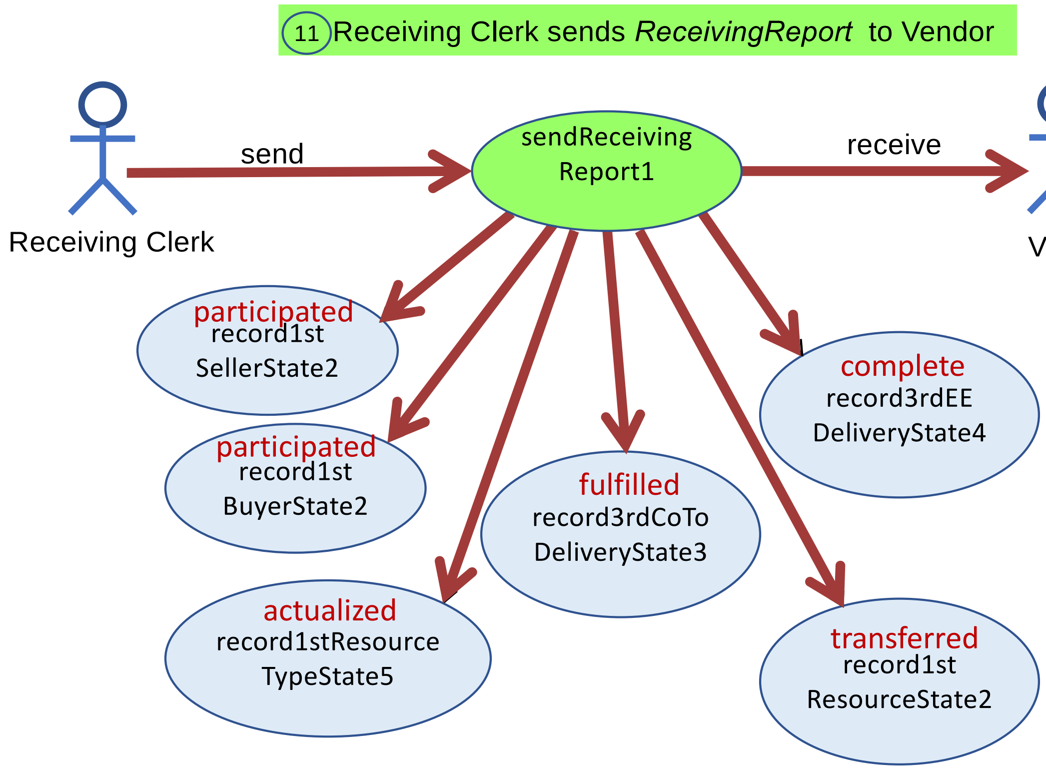


Figure 6-5 – Effect of Business Event on Object States
Source: Horiuchi and McCarthy (2011)

[46] For example, state machines for other resource types might include steps for obtaining funds from a bank in the financing business process or steps for initializing agent types such as hiring internal auditors in the acquisition business process.
[47] The Horiuchi and McCarthy implementation follows the business process example illustrated in Figure 4-19.
[48] Figures 6-5 and 6-6 include Java programming artifacts at both the type (Java class) and the instance (Java object) level, as adapted from the computational proof of concept by Horiuchi and McCarthy (2011). For example in the middle of Figure 6-5, an arrow shows that "an instance of a receiving report happening" would cause "the third commitment's delivery state" to transition from "in-service" to "fulfilled." This same transition is shown as a red check on Figure 6-6 in the fifth column over, the third box down.
[49] This figure expands on step 11 from Figure 4-19 – the receiving clerk indicates resources have been received.

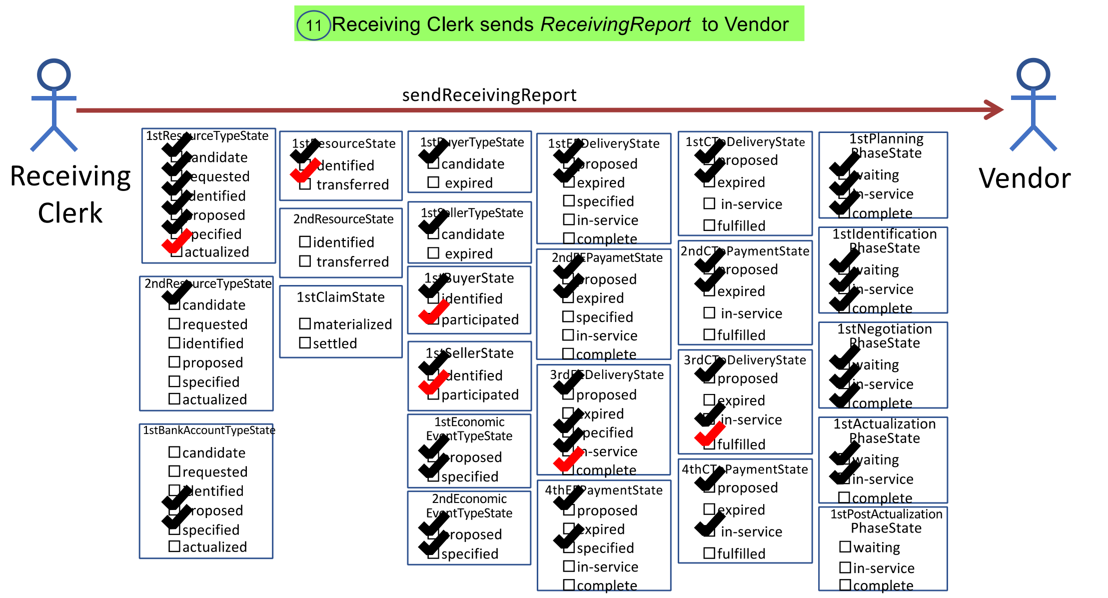


Figure 6-6 – Determining Collaboration Status by Total Effect of Business Events on Object States
Source: Horiuchi and McCarthy (2011)

The Horiuchi and McCarthy (2011) state machine was limited to a simple example that showed no workflow diversions (i.e., no enumeration of exception branches or the "unhappy path"). Horiuchi and Shimizu (2016) expanded their example using the concept of colored Petri Nets to account for a much wider range of workflow possibilities. Additionally, Hunka and Zacek (2015) integrated components of the DEMO ontology with REA in their state machine work.

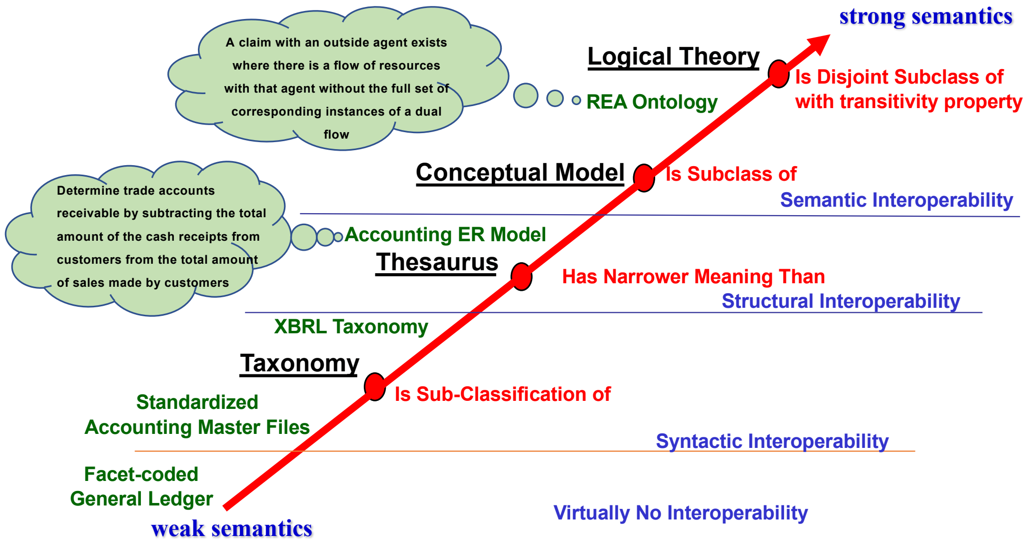
The REA state machine work needs considerable additional development because its conceptual specification is far from complete. Each of the REA ontological categories needs to be vested with an array of possible state life cycles to account for different sets of workflow best practices. This can be done top-down conceptually as evidenced by the Horiuchi and McCarthy (2011) work, but it can also be done bottom-up by taking workflow best practices from the existing enterprise system literature and then molding them to the REA ontological categories: for example how a person is hired and becomes an internal agent of a firm. Initial work on this approach has been started by Boubaker, Leshob, Mili, Charif (2017) who used existing workflows to develop REA models through the use of design patterns. Once such tentative conceptual models are realized, top-down conceptual analysis can proceed with state machine component specification.

Augmented Intensional Reasoning in REA

The ability to reason or "… to discern meaningful patterns … (Dumas & Alexander, 2016, p. 1303)" has been considered critical for intelligent functioning. Therefore, the degree to which the constellations in the REA ontology can support automated or patterned reasoning, can be seen as evidence of the semantic nature of the REA constructs. The most advanced form of automated pattern-matching for the REA ontology was termed augmented intensional reasoning by Geerts and McCarthy (2000a; 1992c), and we turn now to a discussion of future work needed in that arena.

In 2003, Leo Obrst devised an interoperability spectrum for differentiating strong vs weak semantics in information systems (Obrst, 2003). We illustrate that spectrum here in

Figure 6-7 with different kinds of accounting systems. Orbst's scale runs up and to the right in our example from **no interoperability** (facet-coded general ledgers) to **syntactic interoperability** (standardized accounting master files or an XBRL (eXtensible Business Reporting Language) taxonomy) to **structural interoperability** (like an accounting entity-relationship constellation connecting customers to sales, and cash receipts to both customers and sales) to **semantic interoperability** (like an ontology-based application with the REA metamodel stereotypes coded and enforced).



<u>Figure 6-7 – Interoperability Spectrum</u>
(from Leo Obrst (2003), adapted to accounting by authors)

Procedural vs. declarative distinctions have a robust history in the various fields of knowledge representation. Sowa (1984), following Simon (1969), has argued that as knowledge in a domain increases, more of that domain's procedural knowledge can be translated to declarative structures. Geerts and McCarthy (2000) demonstrate such a trade-off in accounting by first defining a procedure for how to arrive at accounts receivable (p. 136):

> Determine trade accounts receivable by subtracting the total amount of the cash receipts from customers from the total amount of sales made by customers.

They then proceed to demonstrate how this accounting concept can be represented at a higher level declaratively once the accounting data model is enhanced with ontological typing (136):[50]

> A claim with an outside agent exists where there is a flow of resources with that agent without the full set of corresponding instances of a dual flow.

These two claim materializations are illustrated along the interoperability spectrum with clouds in Figure 6-7. The former is termed ***intensional reasoning*** by Geerts and McCarthy, while the latter is termed ***augmented intensional reasoning***.

---

[50] The Geerts and McCarthy (2000) reasoning work was first outlined in Geerts and McCarthy (1992), and the ontological typing was done in PROLOG by Geerts (1993).

The ability of any accounting system to support this kind of augmented intensional reasoning depends on its ***epistemological adequacy.***[51] McCarthy (1987) argued that a knowledge-based accounting system is one with, "… the ability to represent definitional features of the environment faithfully." Geerts and McCarthy (2000) extend this idea by saying that for an accounting system to display such adequacy it must support a "…full extent of intensional reasoning in materializing data-dependent conclusions... (p. 137)."

The intensional reasoning work explained above has been extended by others, most notably Ito and Vymětal (2013) who used modal logic with the REA accountability infrastructure. However, much, much more research work remains to be accomplished in extending such automated reasoning to the scheduling, policy, and workflow components of REA and in deriving extensions of business rules for materializing a much larger set of managerial conclusions.

Materializing and Reifying REA Claims

Claims for and against the enterprise were defined by McCarthy (1982, p. 568):

> Claims or future assets as they are called by Ijiri (1975, p. 66-68) derive from imbalances in duality relationships where an enterprise has either:
> (1) gained control of a resource and is now accountable for a future decrement (future negative asset) or
> (2) relinquished control of a resource and is now entitled to a future increment (future positive asset).

In other sections of the original REA paper, there was considerable discussion of claims and methods for deriving them, and that discussion continued in McCarthy (1984), Geerts and McCarthy (1997a), and David, Gerard, and McCarthy (2002). However, in no case was there a resolution to the problem first posed in 1982: which claims should become candidates for base classes in any M1 REA implementation and which other claims should always be relegated to procedural materialization (with M1 applications that range between these extremes to be decided on a case-by-case basis). In this section, we discuss those judgements and make a case for both debt instruments and equity instruments as essential base classes in most implementations. Our work here is predicated on coupling REA debt and equity work with the concepts of another ontology called the Financial Instrument Business Ontology (FIBO). The alignment between these two ontologies is already underway (Bennett 2015), and for our claims work here, we will discard Ijiri's terms and use the much simpler terms espoused by FIBO (Bennett, Gilmore, & Nehmer, 2018): ***rights*** for future positive assets and ***obligations*** for future negative assets.

In Figure 6-8, we recreate a portion of Figure 7[52] from the 1982 REA paper and add to it the FIBO concepts of *rights* and *obligations*, plus one additional ontological category (***claim***) and two ontological associations (***materialize*** and ***settle***) adapted from the ISO 15944-4 (2007) standard. On the left of Figure 6-8, we illustrate an *obligation* or what happens when the revenue process's cash receipts exceed revenue sales to the same customer at a point in time -- we materialize a prepaid revenue obligation if one is needed. On the opposite side of the figure, we do the reverse -- we materialize an accounts-receivable *right*

[51] Epistemological adequacy is a term first used in McCarthy and Hayes (1969).
[52] Like its original, Figure 6-8 is an M1 example from the REA trading partner perspective.

when sale amounts exceed cash receipt amounts (again if needed). In both cases, we see that a ***materialize*** relationship causes a claim to increase, while a ***settle*** relationship causes a claim to decrease.

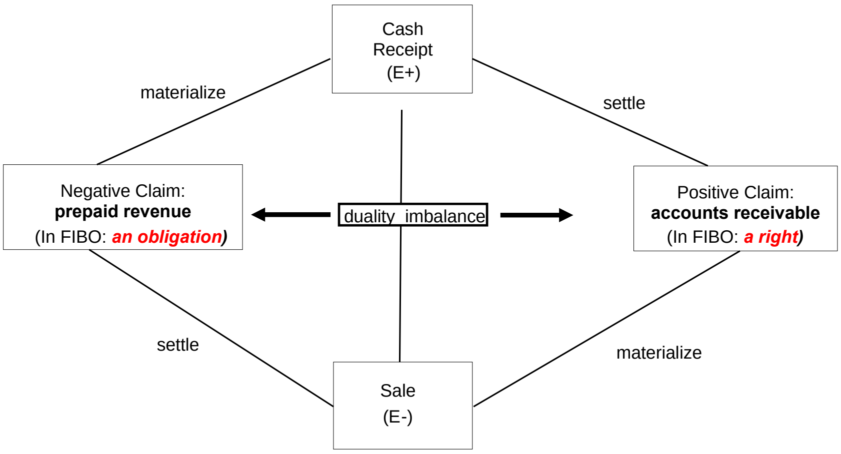


Figure 6-8 – Claims as Imbalances in Duality Associations – a Revenue Process Example (Trading Partner View)

Source: Adapted from McCarthy (1982, p. 568)

In Figure 6-9, we move our discussion of claims to the independent view at the M2 level. Since we cannot always anticipate which of the from-transfers in an exchange will occur first, we will need to use other roles to distinguish the economic events. Hence, we label the first transfer-from agent as the ***party*** and the corresponding transfer-to agent as the ***counterparty***; then we keep those designations consistent for the second transfer-from.[53] This establishes the definition of the REA ***claim*** category and the concomitant definitions of the ***materialize*** and the ***settle*** associations.

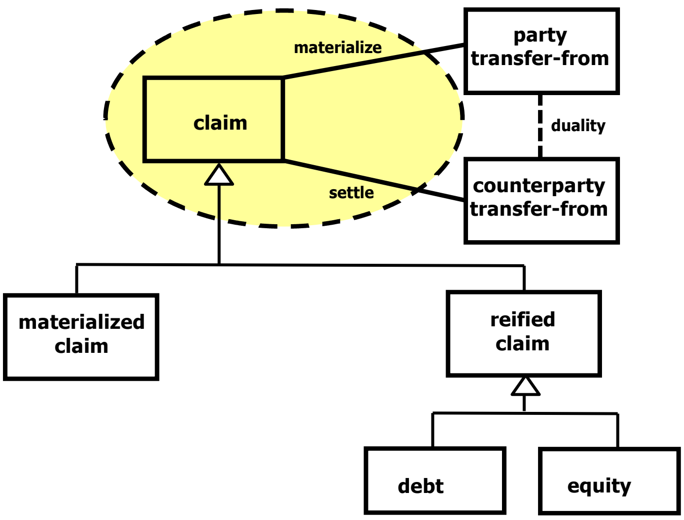


Figure 6-9 – Claims as Imbalances in REA Duality Associations (independent view)

[53] Readers should note that the independent REA metamodel also uses the ideas of party and counterparty to effect a temporal ordering on negotiators for a contract. These two designations are actually independent as one occurs in the negotiation phase and the other in the actualization phase of a business process. Parties to the contract may however stipulate that the party-counterparty designations remain consistent in both phases.

At the bottom of Figure 6-9 another important claim dichotomy for REA (which is conceptually relative (Sowa 1984), but extremely useful) is illustrated. Because of commercial practice and contract law, and as reflected in detailed (but changing and open to interpretation) accounting rules,[54] some claims **must** become base categories in an REA implementation if they are involved in particular business processes. Claims are duality-association imbalances in the REA metamodel of Figure 5-8, but in cases where they must become stand-alone categories, these associations become ***reified*** (or made independently real) in a model. Two strong candidates for such reification are illustrated at the lower right of Figure 6-9: ***debt*** and ***equity***. On the bottom left of Figure 6-9, we list the default case for REA claims which is quite simple: we do not include claims as essential categories in the ontology, but if user needs necessitate their use, we simply use a ***materialization*** process to obtain that particular transaction view.

If equity or debt claims are reified, their inclusion in the ontology may necessitate modeling additional economic events that effect their inflows and outflows. Using the much more familiar trading partner view for displaying what those events may look like, we illustrate in Figures 6-10 and 6-11 what the flows to equity obligations and equity rights might be.[55] Following in Figures 6-12 and 6-13, we portray the same kind of example transaction sets for debt obligations and debt rights.

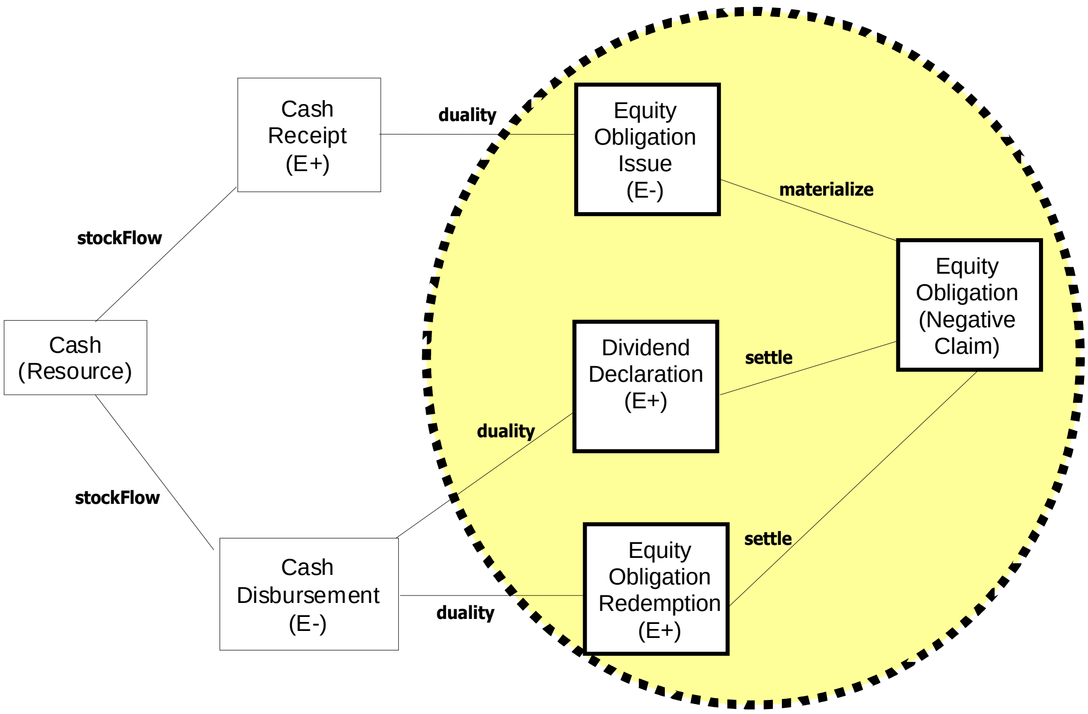


Figure 6-10 – Reifying Equity Obligation in REA (Trading Partner View)
Source: adapted from McCarthy 1984, p. 6 ; David et al. 2002, p.45

[54] See PWC (2017), and FASB Concept Statements 6 (1985) and 8 (2010).
[55] Figures 6-10 and 6-11 model a subset of equity claims as they commonly exist in some "going concern" firms. More fundamentally, equity is a residual claim that might not involve dividends and stock reacquisitions.

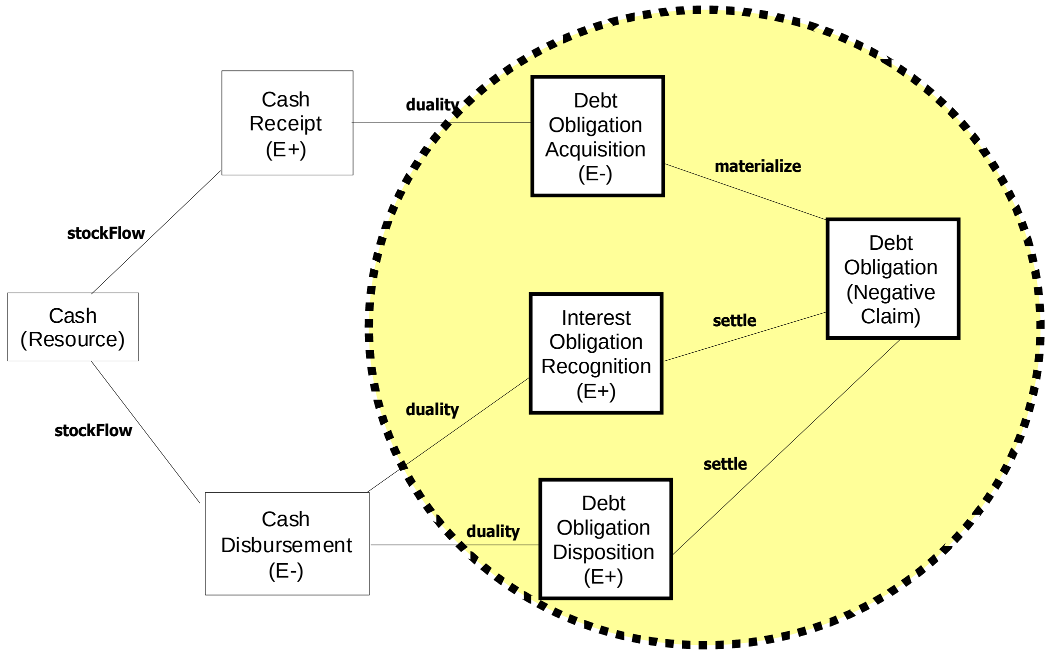


Figure 6-12 – Reifying Debt Obligation in REA (Trading Partner View)

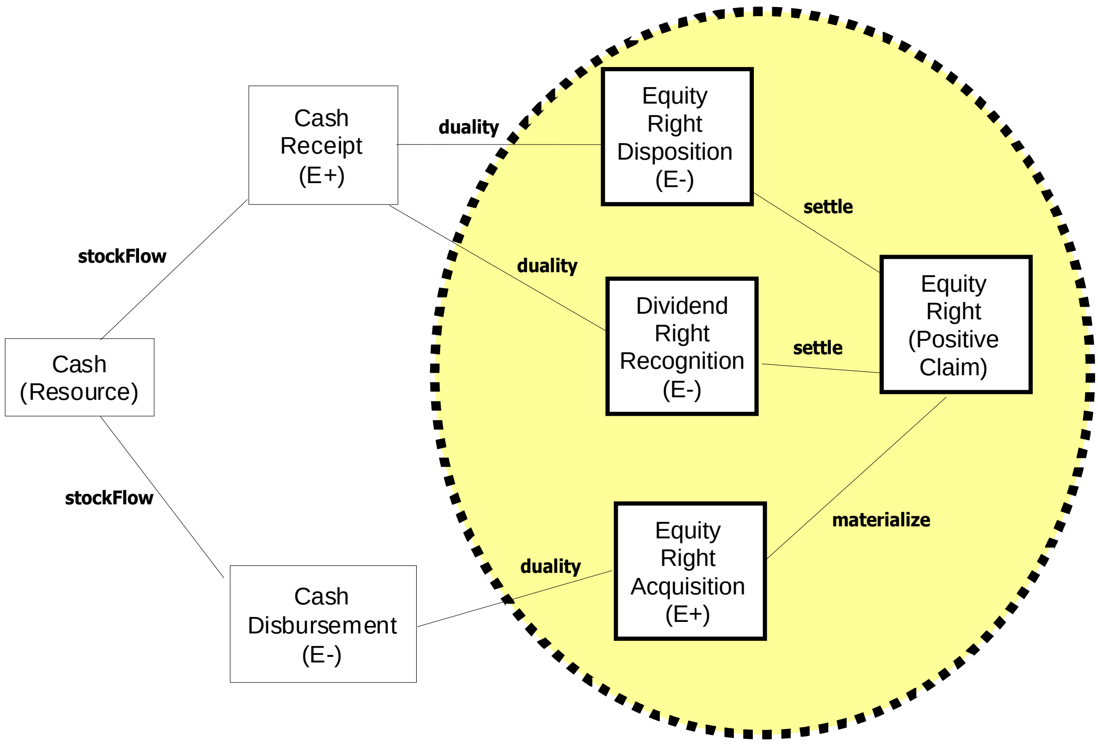


Figure 6-11 – Reifying Equity Right in REA (Trading Partner View)

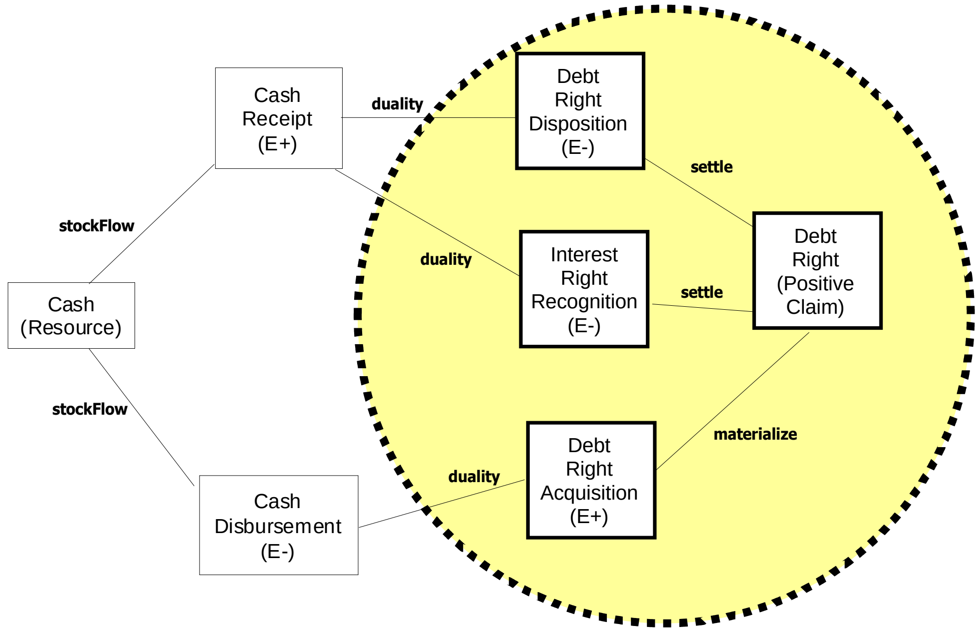


Figure 6-13 – Reifying Debt Right in REA (Trading Partner View)

Future research into REA claim structures is an almost unlimited field. Debt instruments, equity instruments, and other financial structures such as derivatives are awaiting research attention as their complicated and opaque operations remain unfathomably complex and open to wide interpretation.[56] The REA ontology occurrent categories for economic events, commitments, contracts, and business events offer a modeling platform for making the semantics of these instruments much more consistent and transparent. These REA applications will still be quite difficult to formulate correctly, but the ontological framework specified in the prior chapters should make such an undertaking feasible.

Financial Accounting Disclosures

As evidenced by the opening acquisitions examples of chapter 1, traditional double-entry accounting, and REA modeling have very different approaches to the representation of economic phenomena. However, it is certainly true that the ledger structures of Figure 1-2 will still be used in the vast majority of most enterprises, because of the legacy demand for account balances to support financial accounting disclosures. In an unadulterated REA M1 implementation, there is quite simply no need for a declarative general ledger (G/L), but actual implementations may dictate the use of such structures if desired for specific circumstances. In this section we discuss how to materialize G/L balances from REA applications in order to fulfill such needs.

The pioneering work in materializing account balances from REA databases was done by Gal and McCarthy in the early 1980s with two different database systems. The first of these was a network or CODASYL database (Gal & McCarthy, 1980; 1983) called GPLAN which used navigational programming to build a working transaction system and to produce figures like LIFO cost of goods sold, accounts-receivable, and other components of a general

[56] Two research papers that describe initial efforts at integrating REA Semantics with debt and equity modeling are (Bennett, Gilmore, & Nehmer, 2018; Fischer-Pauzenberger, Ondra, & Schwaiger, 2018)

ledger. At the time of this initial work, there were not viable relational database products in general commercial use that could produce the types of REA database structures seen in Figure 1-3. However, starting in 1981, Gal and McCarthy (1982a; 1986) were able to use the QBE (query-by-example) system from IBM to carry such pioneering work much further to the point where an entire trial balance could be materialized with a single set of nested procedures arranged in a hierarchy that mimicked the traditional declarative structures of the "assets = liabilities + owners' equity" equation. McCarthy (1984) consolidated the insights of these two (network and relational) projects into a hierarchy illustrated here as Figure 6-14. Later, this same hierarchy was used as the basis for other proofs-of-concept done by Denna and McCarthy (1987), and David and McCarthy (1992). This materialization work is expanded here in Figure 6-14[57] and augmented with modules using additional ideas (such as use rights from ISO 15944-4 (2007) and materialized/reified claims espoused in earlier parts of this chapter). An important insight from the Gal and McCarthy implementations (1983; 1986)is that the procedure hierarchy shown as Figure 6-14 will often need navigational (one element at a time) programming control structures (sequence, decision, iteration) to augment the specificational (set-oriented) capabilities of relational languages (Tsichritzis & Lochovsky, Data Models, 1982). For example with raw materials costing methods, LIFO, FIFO, and weighted average, can be done with navigational programming, but only weighted average can be done with specificational programming.

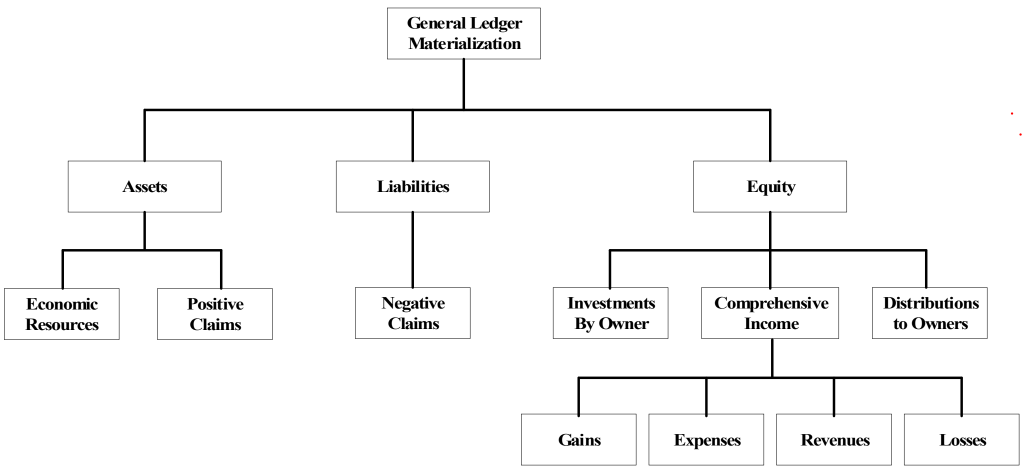


Figure 6-14 – A Procedure Hierarchy for Materializing Account Balances (a General Ledger) from REA Categories
SOURCE: McCarthy (1984, p.9)

A fundamental conclusion about the relevancy of the REA ontology as a sufficient representation of the business domain rests on its ability to support traditional financial accounting disclosures, and the procedure hierarchy of Figure 6-15 provides a platform for exploring the implementation of industrial-strength general ledger structures if such structures are deemed necessary within an enterprise implementation. The materialization of such "minimal general ledgers" is clearly an important topic for future REA research as its

[57] The procedure hierarchy shown as Figure 6-15 is deliberately abbreviated for simplicity purposes. Each of the modules will be expanded further in an actual implementation just as done in Gal and McCarthy (1986). For example, the ***goods*** component could include materializations for raw-materials, work-in-process, and finished goods, among others.

conceptual ideas move toward an alignment with current accounting software. We would contend that the REA ontology poses a standard toward which current software should evolve, but this contention needs empirical analysis and validation. This is not a trivial test as it provides a basis for evaluating the completeness of the ontology and its faithful representation of the business domain in its ability to support varied uses or views of its declarative elements, as outlined in McCarthy (1982).

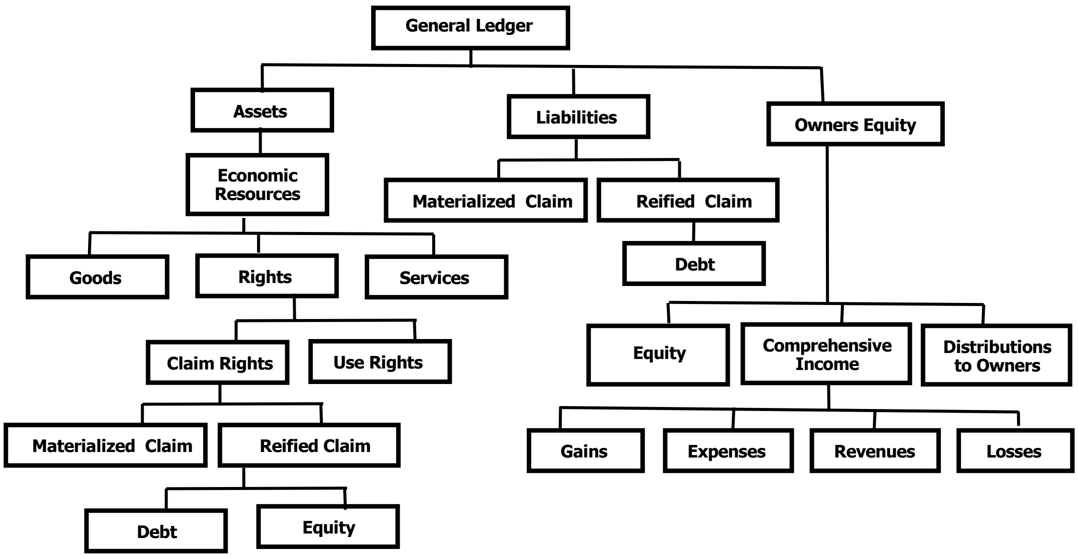


Figure 6-15 – Procedure Hierarchy for Materializing a General Ledger (Trading Partner view)

Managerial Costing and Reporting

As mentioned in Chapter 4, REA modeling of managerial accounting and conversion cost structures has been studied extensively, the earliest examples of which were QBE prototypes by Fedorowicz and McCarthy (1983) and Armitage (1985). Armitage in particular, showed how relational implementations of REA could produce many cost accounting reports and accommodate many different types of costing schemes. However, in our discussion of managerial costing and reporting here, we will rely on the meronymic conversion foundations developed in chapter 4. We will also not be examining specific costing schemes, such as activity-based, absorption, and periodic costing. Instead we intend to focus on a more recent and more conceptual approach to costing published by the IMA (White & Clinton, 2014). The IMA report eschews specific costing scheme discussions in favor of analysis of costing principles such as traceability, attributability, integrated data orientation, etc. In the paragraphs that follow, we speculate on how some of those principles can be accommodated with REA structures.

In an earlier IMA report (The Institute of Management Accountants, 2005) the history of creating different data and the inappropriateness of the practice of keeping information for different decisions separated with respect to managerial costing was observed:

> "….. [managerial costing has been] an area that has often been considered the Wild West of the accounting profession because of adages such as "different costs for different purposes," "different costs for different questions,"" relevancy is all that matters," "use what works for your company," and so on. Beliefs and pseudotruisms

such as these are often used as an excuse or reason not to pursue a deeper analysis of the foundational principles and concepts that underlie managerial costing analyses and models (p.7)."

Following this logic, the 2014 IMA report specifically rejects the notions of different costs for different reports and instead adopts the idea that a conceptual framework should be a causal representation of an enterprise's operations – an ideal that puts it squarely compatible with REA modeling. It is also the case that the 2014 report stresses repeatedly the dysfunctional habit of primary reliance on general ledger figures for costing, figures which it notes are often distorted by financial accounting convention. In the illustrations that follow, we explore research horizons at the intersection of the costing conceptual framework and REA modeling.

Using Data Abstraction (Grouping) To Avoid Arbitrary Allocation

Figure 6-16 portrays one of costing's biggest traceability issues: what to do with indirect costs which are resource inputs that are not directly traceable to the primary consumption events in a business process. Different costing schemes handle this issue with differing levels of success, often with heavy reliance on account balances to configure the distribution. The IMA report notes that these costs are often best assigned to ***business levels***, which are aggregations like a product, a product group, a plant, a region, a distribution channel, or even a whole organization.

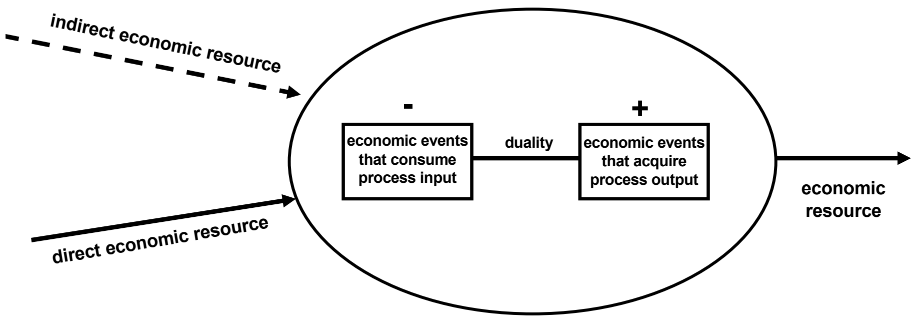


Figure 6-16 – Direct and Indirect Inputs to a Business Process

With REA modeling, these aggregate resources are best modeled with *grouping*, an abstraction mechanism explained in chapter three. As Figure 6-17 illustrates, this grouping can be done either ***temporally*** by using process hours, process weeks, etc., or ***sectionally*** by using work stations, departments, etc. Additionally, this grouping can occur *within* a single process or across multiple processes. Examples of each of these applications are explained next.

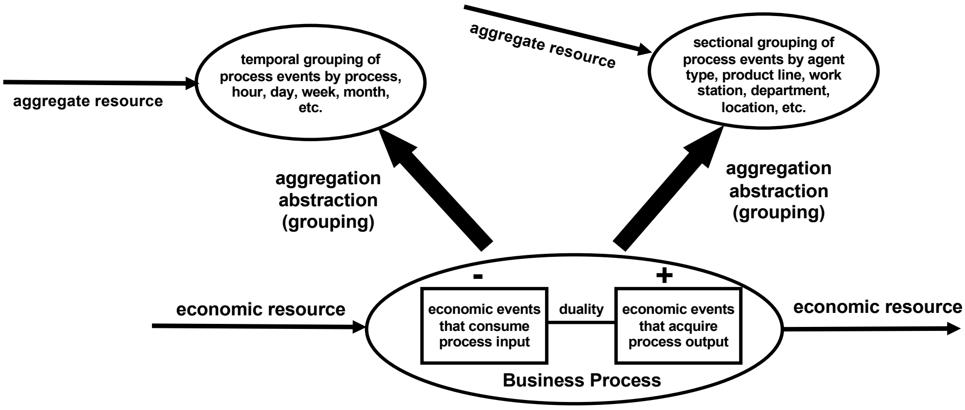


Figure 6-17 – How REA Accounting Solves Process Granularity Mismatches with Grouping

Figure 6-18 is a partial recreation of the *Alaskan Aircraft Expeditions* revenue cycle originally shown in Figure 4-22. This example illustrates grouping for an aggregate resource ***within*** a business process. The aggregate resource is Advertising Service which is assigned not to the primary events of the process but to two aggregations: (1) *Client Segment* which is a grouping of *Customer*, and (2) *Expedition Type* which is a grouping (typing) of *Expedition*. The mirrored reality is that advertising is certainly a component of the combination of goods and services rendered to the customer on an expedition (that is, it is part of the bundle of attributes of interest to the customer), but it cannot be traced directly. The example also assumes that advertising was, in principle, aimed at the grouped aggregates in a causal fashion. We use the association name "inConcert" to illustrate that the consumption of the aggregate does not follow the consumption of direct items like the aircraft, the labor, and the supply items, but it does follow a similar path of use, albeit at a different level of granularity.

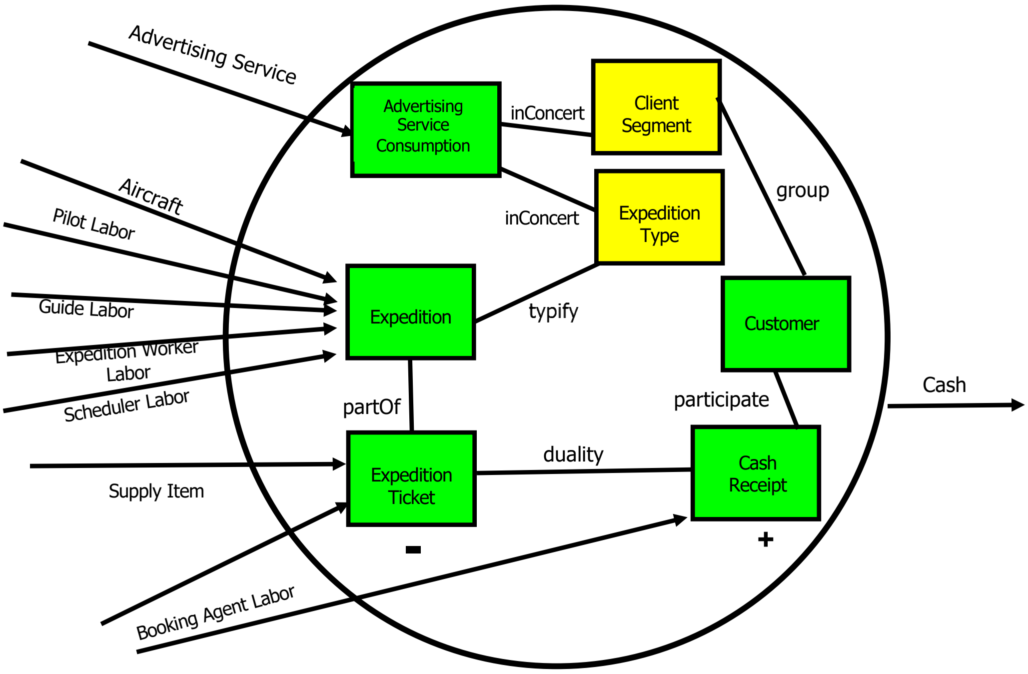


Figure 6-18 – Within Process Groupings for Aggregate Resource

Figure 6-19 (which mirrors an REA business process hierarchical structure first explained in Figure 2-3) illustrates in general how aggregation abstraction (i.e., groupings) can solve indirect costing ***across*** individual business processes. First, a grouping variable for the lower level processes would need to be defined; this grouping could be natural (like the customer segment shown above) or somewhat arbitrary. Then sectional or temporal aggregations could be assigned to the different group instances, and those instances could be aggregated to partitions of the higher level process shown above it. Using the ideas of the IMA report: grouping is a way to effect a combination of specific managerial objectives for which specific aggregate optimization activities are taken.

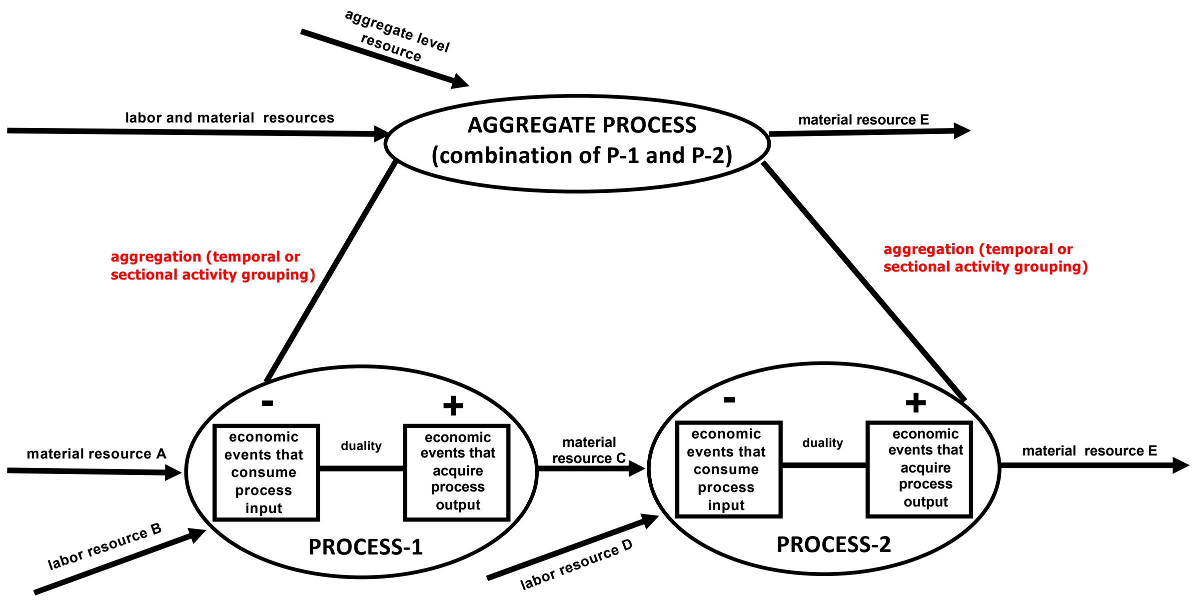


Figure 6-19 – Process Hierarchy Aggregating across Individual Business Processes

Figure 6-20 is another example taken from the *Alaskan Aircraft Expeditions (AAE)* example of Chapter 4; more specifically it is an aggregation across three processes shown on the value chain of Figure 4-21: the aircraft acquisition process, the maintenance service acquisition process, and the fuel acquisition process. To illustrate a managerial costing issue, we augment the description of AAE's operations to include the addition of a supervisor who manages all three processes. This creates a process hierarchy for this segment as shown, where each of the three sub-processes would be aggregated with a grouping class, and then those groups would be assigned to a project (i.e., a grouping) in the higher level process. The supervisor labor would be linked then to the aggregate project via the labor consumption event and a duality association. This modeled reality might, under certain circumstances, be subjected to a traditional costing procedure (as a database view), if managerial and financial users differ on the accepted methods of allocation, but the IMA report clearly discourages such divergence. The important point for the REA modeling is that it reflects the operational reality as closely as possible.

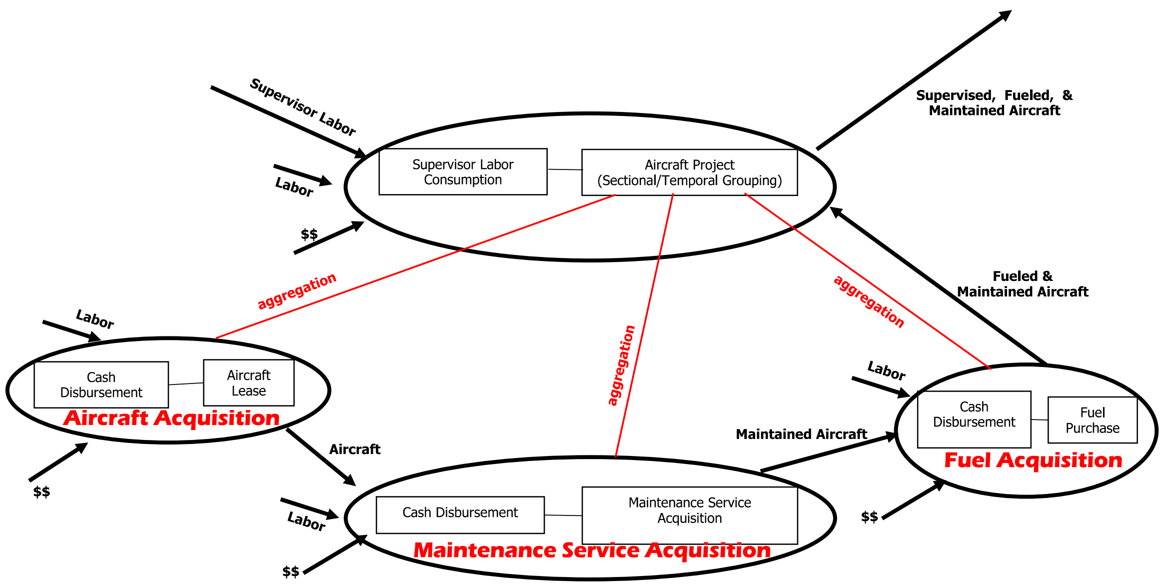


Figure 6-20 – Process Hierarchy Aggregation for Supervisor Labor Input

Aggregating Business Events to the Business Process Level

As illustrated in Figure 4-8, business events may have a hierarchical structure and they may have ordering.

1. As defined in ISO 15944-4, any business events that have duration (that is, they are non-instantaneous) may under certain computing conditions necessitate instantaneous start and finish events, plus there may be other circumstances where nesting makes sense.
2. State machine mechanics may actually provide an ordering for business events, but in the absence of such mechanics and as a matter of business policy, workflow events are usually ordered. In Geerts and McCarthy (2001), policy structures for developing such orderings were specified as ***recipes***.

Regardless of whether business events have duration or order though, there is a need in managerial accounting analysis for relating their occurrence and cost structures to the business process level. As illustrated in Figure 6-21 with the AAE conversion example from Figure 4-29 and Figure 4-30, this can also be accomplished by aggregation where the entire business process instance (in ISO 15944-4 terms, this is a business transaction object) is viewed as a group to which the workflow elements are linked in the aggregate. This linkage causes a number of interesting managerial issues addressed in the IMA (2012) report under the heading of ***attributability*** (pp.52-54), and in some cases, it has intensional reasoning repercussions for REA modeling, so further research here is clearly needed around these discussion points.

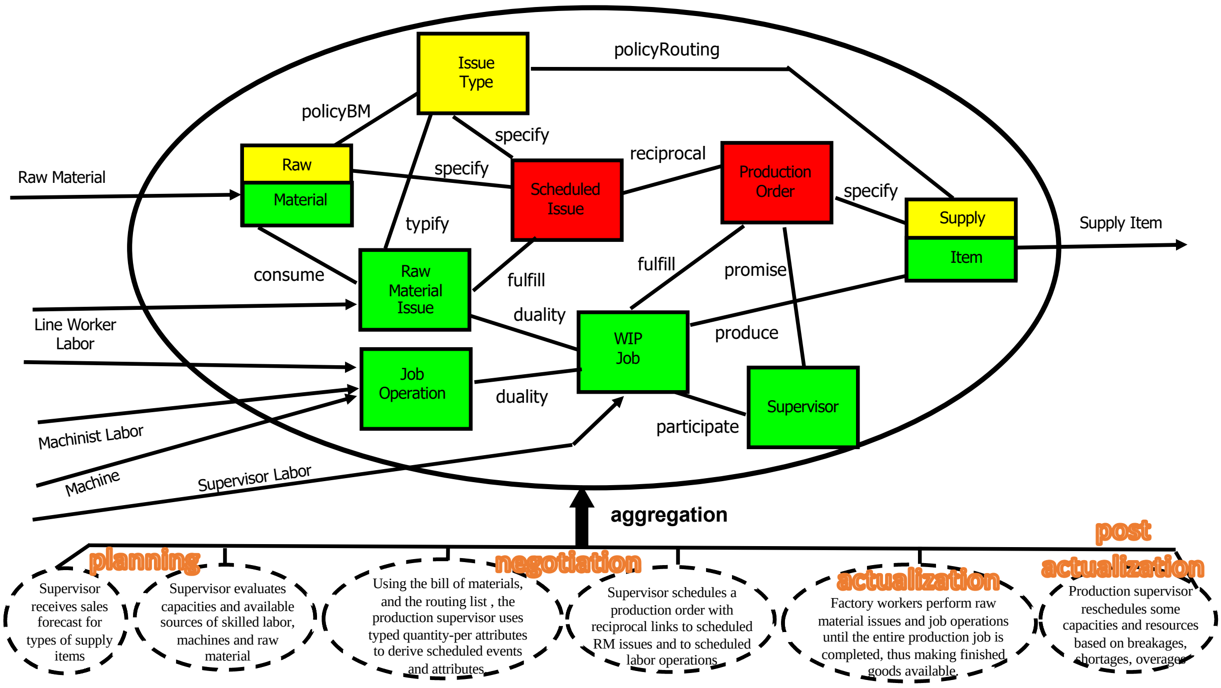


Figure 6-21 – Aggregating Conversion Cycle Business Events

1. As discussed in Geerts and McCarthy (2001 pp. 101-103), there will be occasions when business events rise to the level of economic events because they produce an identifiable acquired resource, in which case that business event leads to a separately modeled business process.
2.
3. The conversion process shown in Figure 6-21 is assumed to be successful (a “happy path” instance) because the planning and negotiation events lead to an actualization instance where a production job took place. In actual practice, there will clearly be “unhappy path” occurrences of the business process where the planning/negotiation is abandoned without actualization increments. These abandoned process instances must be modeled and not arbitrarily allocated to successful instances, because this distorts decision-making information. For this, the IMA report says resource consumption events “which cannot be quantitatively associated with specific outputs in a causal manner fall within the sphere of attributability. (White & Clinton, 2014, p. 53).” In REA terms, these abandoned instances can be linked to higher *business levels* in the manner illustrated here in Figure 6-18 (within process) or Figures 6-19 and 6-20 (across processes).

The difficult issues identified above are just a few of the ways in which REA modeling can be used for theoretical accounting advances. Using the declarative and procedural features of the ontology to construct a platform for the effective implementation and use of many features of the conceptual framework for managerial costing will obviously become a fruitful area for future research by managerial accountants.

The next section looks at another critical informational issue for a business ontology; the ability to deal with the management restrictions and policies for the execution of the organization’s activities. These restrictions and policies fall under the general heading of internal controls and are included under the more general control of all risks to the organization. The Committee of Sponsoring Organizations of the Treadway Commission (commonly referred to as COSO) has developed and updated a framework which defines internal controls in business organizations (2004; 2013; 2016).

Internal Controls

In previous sections, we demonstrate how the declarative or intensional components of the REA ontology can be used to create views which describe accounting and other business phenomena. General ledger accounts, business claims, value networks, etc. can all be considered as views generated from REA classes and associations. This section looks at internal controls, which are used to make judgements about the state of the business. Auditors, must consider data from an accounting information system to conclude whether the organization is either well controlled or has a material internal control weakness (United States Congress, 2002). This is similar to the conclusion required by auditors concerning whether accounts contain a material misstatement. Thus, there are two issues that will be addressed in this section: the relationship of REA's declarative components with internal controls, and how these information system components might be used to materialize a conclusion about the quality of a firm's internal controls.

In the COSO framework (1992; 2004; 2013), risks to the firm relate to its inability to meet objectives in four areas; strategic, operational, reporting, and compliance.[58] To ensure the firm can meet these objectives, COSO describes an internal control system having five components: control environment, risk assessment, control activities, information and communication, and monitoring (2013). In previous sections of this chapter, the completeness of the REA ontology was evaluated by its ability to provide relevant reports, thus fulfilling a component of COSO's reporting objective. These reports concern the state of business processes (state machine mechanics) and the creation of information relevant for external and internal stakeholders (financial and managerial disclosures). Here we will examine how previously described components of the REA ontology provide information relating to operational objectives and support monitoring activities as they relate to certain firm risks.

Reporting objectives are central to the functioning of a firm and include the ability of its information system to provide accurate and complete financial and non-financial information. An information system must also be able to provide information about the presence and functioning of internal controls. The comprehensiveness of the information system can be evaluated by its ability to provide information appropriate for monitoring the progress of the firm in meeting operational objectives (COSO, 2004, 2013). So as employees perform business events, prescribed by management, there should be sufficient information reported to management, or any stakeholder, to ascertain whether these activities progress the firm toward meeting operational objectives and whether the firm complies with regulatory requirements. Thus, a well-controlled company has an information systems which captures the relevant business events, restricts or controls these business events, and provides relevant stakeholders with information that will allow them to monitor the business events (COSO, 2008; 2009). There are many factors that impact meeting these internal control objectives; some of which can be controlled while for others controls are more difficult. Because of its ability to represent the operations or business processes of the firm, the REA ontology supports the representation of various controls necessary to restrict and then to monitor the business processes (Gal & McCarthy, 1985a; 1991). However, business processes operate in

[58] In the 2013 edition of COSO's report, strategic objectives were moved to Enterprise Risk Management (COSO, 2016)

the broader organizational environment. Thus, there can be factors in this environment which might have a pervasive effect on controls and therefore risks across all business processes. The next section will examine controls in the business environment and discuss how they may be considered within the REA ontology.

Control Environment

The control environment includes such factors as management's risk philosophy, integrity, commitment to competence, assignment of authority and responsibility, and organizational structure (COSO, 2004, 2013).[59] The factors in the control environment are seen to affect the general functioning of controls within the firm as opposed to affecting any specific control such as the design and the function of a business event (COSO, 1992). For example, the specification of the management's risk philosophy, "… is the set of shared beliefs and attitudes characterizing how the entity considers risk… (COSO, 2004, p. 27)." This risk philosophy can impact many activities of the firm indirectly as management at one firm might allow more risky operations than at other firms. A firm's risk philosophy would be difficult to represent directly in REA's intensional components. However, it might be possible to infer risk philosophy by examining a large set of board directives provided to management. While this component of the "tone at the top (COSO, 2013)" is difficult to describe in terms of system attributes, there are other factors in the control environment that can be represented directly within the REA ontology's intentional features.

Competency is the ability to appropriately perform particular business events within the organization. It is reflected by, "…the knowledge and skills needed to perform assigned tasks (COSO, 2004, p. 31)." The commitment to competence is evidenced by two actions of management. First, is whether management has determined how each business event needs to be performed and then specified the skills needed to perform those tasks at that level (COSO, 2013). Thus a prerequisite to evaluating control adequacy is an understanding of the effort management made to determine the level of competency required for specific business events.[60] The second piece of evidence to determine commitment to competence is the degree to which employees are actually assigned to tasks for which they have management's view of the requisite level of competence as indicated by management's specification of its necessary skills.[61] Considering the perspective of Auditing Standards the process of evaluating internal controls over employee competency consists of two components enumerated in Auditing Statement 5 (Public Company Accounting Oversight Board, 2007). First, whether a policy has been formulated that indicates which employee types should perform which business event (para 42-43), and second is whether the correct employees are actually deployed to these tasks (para 44-45). AS5's requirement to verify both the design of components of the processes, such as determining competency, and examination of the actual hiring procedures can be directly connected to REA policy and accountability layers.

---

[59] The 2013 framework (COSO 2013) includes additional principles such as independence of the board from management and that management exercises oversight with respect to the assignment of authority and responsibility.

[60] There are activities management can take to reduce the impact of unqualified employees. The assignment of employees that do not meet the management preferred characteristics for an employee type assigned to a particular activity can be mitigated by the application of supervision to less than competent employees.

[61] COBIT 5 (ISACA , 2012) management practice AP007.03 combines both the definition of skills along with verification that personnel have the appropriate level of competencies.

The Policy Layer of Figure 3-16 shows the relationships between type images as policies. The competence level includes the knowledge and skills which can be included in the specification of employee types. Figure 6-22 expands the policy class, "Employee Types" into instances of employee types that management considers necessary to perform certain business events in the firm. The attributes for "Employee Type" are an indication of the attributes management considers relevant for determining an employee's level of competence, and the quality of the process to determine what a competent Internal Auditor, for example, looks like. By specifying the range of values for specific instances of employee types, the competence level is described. For example, an Internal Auditor needs a BA in accounting, a CIA certification, and has IA training, while a Salesperson needs either a BS or BA, no certification, and has sales training.[62] This specification indicates management's understanding of the knowledge and skills required for certain types of employees. When individual employees are assigned to these types, this information can be used to evaluate human resource standards – how close does this employee match the competence and skill level required. Therefore, policies established by management on required characteristics for Employee Types can be described within REA's policy layer, and can provide information so the competence of employees can be monitored. However, a set of procedures is needed to evaluate whether the control over hiring competent employees is functioning.

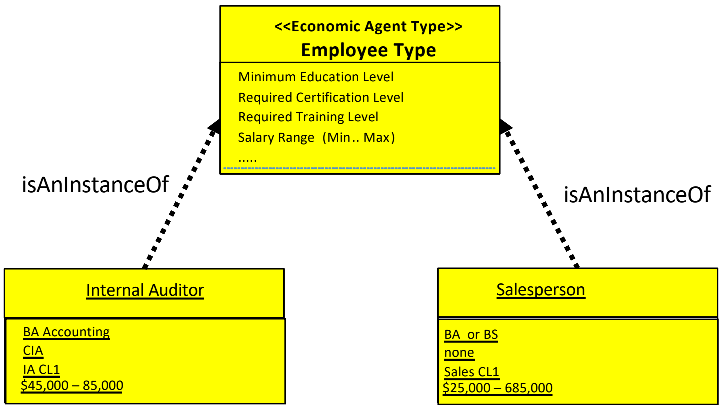


Figure 6-22 – Two Instances of Employee Type

These two factors -- commitment to competence and human resource standards -- both impact the control environment. The quality of the control environment is not limited to any specific activities or objectives, but can have a pervasive effect on the operation of the firm. Competent employees indicate a better chance that business events will be performed correctly, but this does not mean they will. Therefore, control environment factors can affect the ability to meet objectives in these broad areas: strategic, operational, reporting, and compliance, and they may give assurance in areas not part of the review. The next section will look at controls and risk as they relate to the events with direct impact on the performance of activities and meeting objectives.

[62] For a more detailed description of the connections among data models, controls and audits, see Gal (2015)

Event Identification and Control Activities

Other areas in the COSO framework for Enterprise Risk Management (COSO, 2004) require the identification of critical events; those which have a major impact on the firm's ability to meet objectives. Once identified, the firm can enumerate the potential risks, and then determine the appropriate responses to these risks. The REA ontology supports the enumeration of the relevant business events, their interrelationships within business processes, and their effect on business process phases. When management specifies policies concerning how actual processes and events should unfold, this satisfies the COSO requirement of identifying events which are critical to meet operational objectives. The ***typify*** associations between the Policy and Accountability layers connect, "What could or should be (Figure 3-1)" with "What has occurred" and allows management to determine whether policies for execution of business events are followed and whether they support the requirement that controls are monitored (COSO, 2013). Creating schedules (or contracts) which adhere to management's established policies (the ***specify*** associations which connect the Scheduling Layer with the Policy layer) further supports the requirement for effective control over transactions (Public Company Accounting Oversight Board, 2007). The connection of these Policies and Schedules to business processes and phases can allow for comparison of the way in which business events across the firm should be executed with their actual completion. This analysis is a key component of the requirements of Section 303 of the Sarbanes-Oxley legislation (United States Congress, 2002).

Process flow analysis (COSO, 2004, Exhibit 4.1) requires the identification of, "… the combination of inputs, tasks, responsibilities, and outputs that combine to form a process." The REA ontology views Business Processes similarly. It groups business events into business processes and phases, and therefore the process flows within the organization. Thus, COSO's process flow analysis can be accomplished directly with the ontology's constructs as represented in Figure 3-16 and elsewhere in this monograph. To monitor controls, the analysis of these process flows requires an understanding of the factors that could affect these processes and thus adversely impact the achievement of operational objectives (COSO, 2009; 2008).

Some of this analysis can be conducted at the process level as opposed to the review of instances of business events in the accountability layer. For example, in the revenue business process, if management's policy, their design, for the business events included in the credit verification process are designed to occur after the business events steps for the shipment of the merchandise process, then a conclusion can be made that there is an internal control weakness without reviewing actual business events. From a COSO perspective, if the policy for credit approval is designed incorrectly, then a relevant risk related to the reporting of accounts receivable exists. Gal and McCarthy demonstrated how these controls can be applied to these processes or views of a business process (Gal & McCarthy, 1982b; 1991) (Figure 2-3). If the policy for design of processes is correctly followed, (i.e. shipment events are designed to occur after the credit verification events), then events at the Accountability Layer need to be reviewed to determine whether the actual credit verification is done according to Policy Layer descriptions. From the COSO perspective, this review of controls should be done on the design of processes (Public Company Accounting Oversight Board, 2007, pp. para 42-43), and also on the operation of the process (Public Company Accounting Oversight Board, 2007, pp. para 44-45).

By connecting the policy, scheduling, and accountability layers across business processes within the firm, management can monitor adherence to internal controls in specific processes within the firm that have a critical (or material) impact on meeting operational objectives (Committee of Sponsoring Organizations of the Treadway Commission, 2009; 2008). Gal and McCarthy (1991) show how the semantic specification of a business process, which would include a representation of how the actual business events should proceed (through policies and schedules), can be used to automatically enforce and evaluate operation of internal controls in the firm. However, overall internal control evaluation requires a judgement about which areas present the greatest risk as related to their preferred strategy (COSO, 2016).

Only after the risks on the execution of processes have been identified, can control activities be developed. These activities include: top-level reviews, direct functional or activity management, information processing, physical controls, performance indicators, and segregation of duties (COSO, 2004, Exhibit 7.1). Some of these control activities require examination of actual events and make use of the managerial costing capabilities already presented. For instance, top-level reviews include the comparison of actual performance versus budgets or standards. Additionally, activity management includes reviews at a more micro-level such as reviewing daily cash flows. Each of these control activities make use of information about certain intensional components of the ontology presented in previous sections. A final control activity, discussed in this section, which can be formally modeled in the REA declarations is segregation of duties.

During the Actualization phase of each business process there is an increment event paired with a decrement event along with business events to accomplish these events.[63] The state machine mechanics include a connection between these business events and completion of the business process phases. In the specification of the Event Types, there is a policy association with an Employee Type. Management's assignment of an Employee Type to each of these events that make up the Negotiation phase (in figure 4-17) is an indication of their assessment of the skills and knowledge required to perform the activity. The policy that assigns the employee types to specific organizational units is an indication of management's design of the operational policies relating to segregation of duties (see Figure 1-13 as an example). For instance, in the actualization phase of the revenue business process, this could be accomplished by segregating responsibility for the increment and decrement business events. By assigning the Employee Type "Cashier" to the Event Type "receive cash", and assigning Employee Type "Salesperson" to the to the Event Type "Sale", this design of system policies indicates functions that would be considered incompatible and thus should segregated (Gal & McCarthy, 1985b).

Throughout this section, distinctions have been made between design and operational controls. This difference between designs or policies related to business events allows for evaluations of management's efforts to delineate Policies and the Monitoring (or evaluation) of segregating duties in particular instances (Gal & McCarthy, 1985b). While authoritative

[63] In a functioning system management can make a determination about which business events to control and those business events for which no formal control is needed. For instance, at one point in time, it may be sufficient to assign the activity for credit check to a manager and just look for an indication that the activity was performed. At a later date management might determine that to properly do a check credit, three sources must be checked. The structure of the REA ontology allows for addition of business events and therefore management might need to extend the current controls. See (Gal, Geerts, & McCarthy, 2009) for a more complete discussion of these issues and the impact on enforcement of restrictions.

literature (Rittenberg & Schwieger, 2001) indicates which types of duties should be segregated, management could make a determination that certain activities do not need the level of segregation implied by COSO. The actual conclusion about the adequacy of the segregation, or for that matter any control, could be influenced by factors such as the presence of additional supervision or the potential impact on the firm's financial statements.

Internal controls can be thought of as restriction of business events so that firm's objectives have the best chance of being achieved. Within frameworks such as COSO there is strong emphasis on understanding and evaluating the design of a firm's information system prior to understanding and evaluating its operations. Representing business events in terms of a general ledger doesn't provide any indication of the adequacy of the information system's design nor its actual operation. This is a further reminder of a limitation of traditional accounting systems expressed in the preface to this monograph. In contrast, the representation of a firm's information in terms of the REA ontology's components naturally maps to the many of the terms of the COSO framework's concepts and therefore the required connection to internal controls and their evaluation.

Some Further REA Design Extensions

This chapter has highlighted some needed extensions to REA, but it has concentrated on extensions (like state machines, intensional reasoning, and financial instruments modeling) that have their base work started in at least preliminary fashion. However, there are other possible projects waiting behind these for concerted design efforts. We mention some of these below in very preliminary fashion.

- The role of standardized attributes (like amount) and attribute bundles (like address) in REA ontology construction and use – Attributes were first introduced in Figures 1-6 and 1-7, and then they were used as examples in succeeding chapters. This work needs much more development and discipline, as attribute standardization is very important in REA uses like intensional reasoning. We intend to follow up on this need, concentrating primarily on the UN-CEFACT work on Core Components (United Nations Centre for Trade Facilitation and Electronic Business, 2009) and the OASIS developments in the Universal Business Language (Organization for the Advancement of Structured Infomation Standards, 2013).
- The exploration and possible expansion of the base set of REA categories and their associated relationships – from Geerts and McCarthy (2000b), a high priority item for the authors will be the integration of business events, business event types, and recipes which can be used to specify best practices in workflow. Most important for this integration, the enumeration of workflow recipes must be integrated with the various normative life cycles of business transaction entity states of the type illustrated in Figure 6-1. Other possible entity expansions for REA are mentioned in the ISO 15944-4 specification, like the **Business Location** and **Location Type** business transaction entities illustrated with darker shading in Figure 6-23. And finally (from a combination of Geerts and McCarthy (2000b) and ISO 15944-4 (2007)), possible supplementary relationships for REA might need to be considered as informative additions like **economic control**, **linkage,** and **custody**. For expansion of REA in general however, attaching new components is an effort that must be approached with caution as a minimal normative and informative set will usually be the best specification.

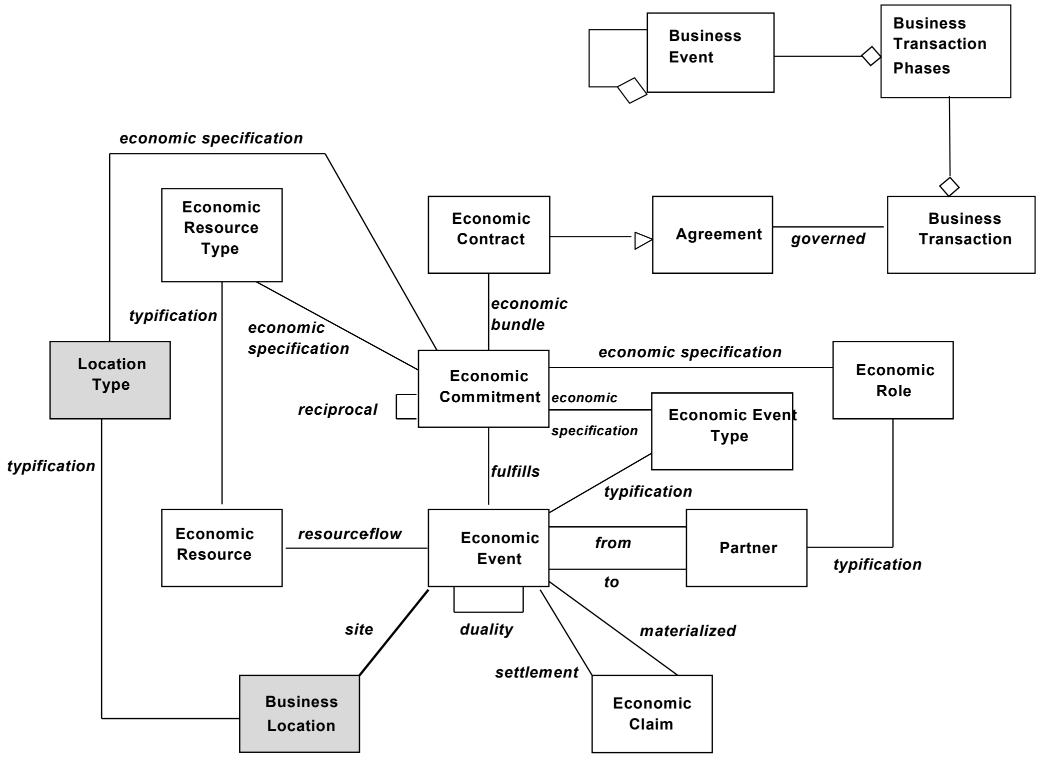


Figure 6-23 – Classes for Future REA Ontology work Source: ISO 15944-4 (2007)

- The exploration of non-binary or mediated business processes – REA as explained in this monograph concentrates primarily on binary exchanges between two agents with competing economic interests. There are many business processes where third (and fourth and higher) parties play significant roles that must be specified. Such parties include for example banks, logistics operators, and government agencies. A preliminary approach to such a specification might attempt to break the mediated exchange down into its binary components which can then be aggregated (McCarthy W. E., 2003c). However, this initial direction needs much more concentrated study.

Conclusion

In its simplest form, Occam's razor states this for the explanation of a model:

Entities should not be multiplied unnecessarily[64]

We have followed this advice as closely as possible in demonstrating and explaining the classes (i.e., the categories) for the REA ontology. As a summary of REA, we include here two class diagrams as Figures 6-24 and 6-25:

[64] https://www.merriam-webster.com/dictionary/Occams razor

Figure 6-24 shows the independent view of the REA metamodel from which the less-general trading partner view may be derived. This is our most important specification, built as parsimoniously as possible.

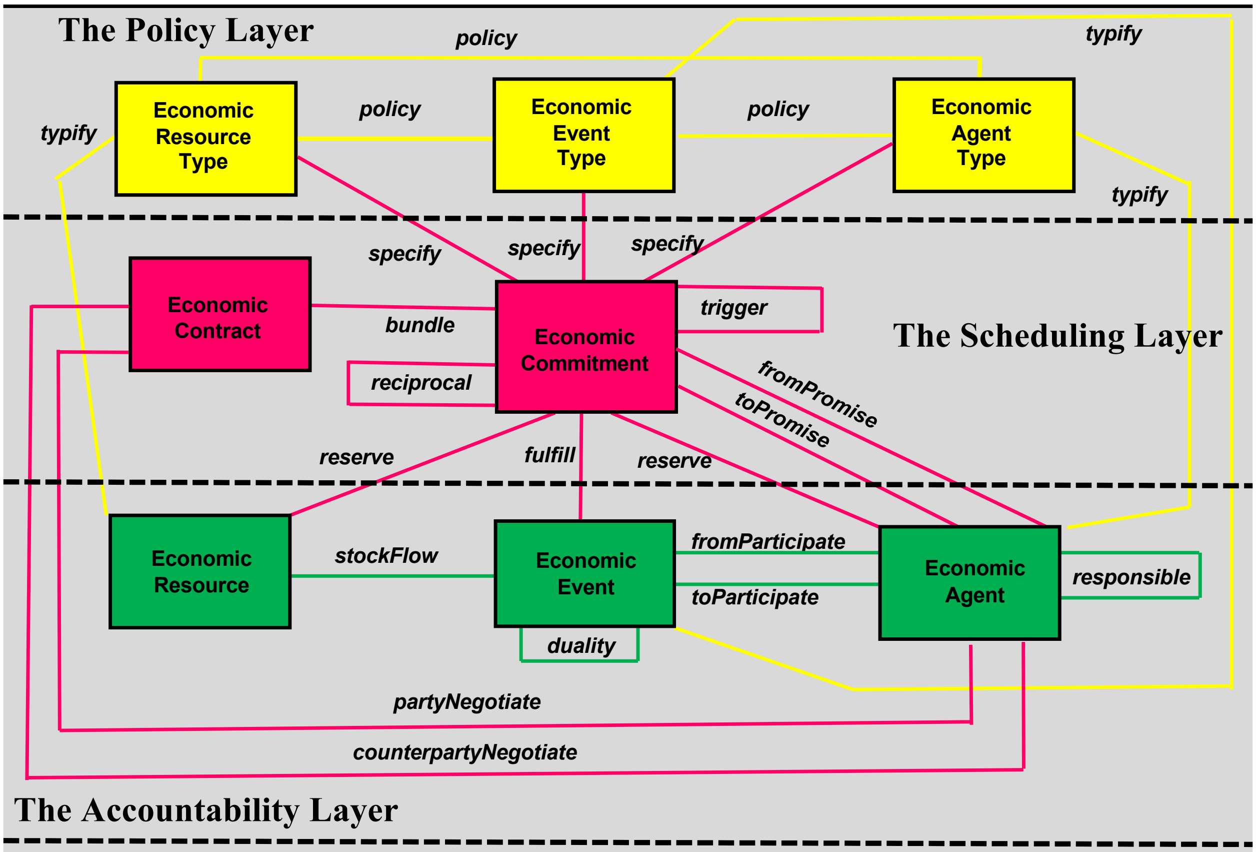


Figure 6-24 – The REA Metamodel (Independent View)

Figure 6-25 shows the REA granularity levels, again in minimal fashion as we have made ***value networks*** and ***value chains*** conceptually congruent as they indeed become for the independent view. Value chains can be derived from the nodes of value networks by process decomposition and Coasian analysis as illustrated in Figure 4-10. We realize of course that enterprise value chains usually evolve organically from existing structures as opposed to top-down analysis of a single production function, but again our purpose here is parsimony.

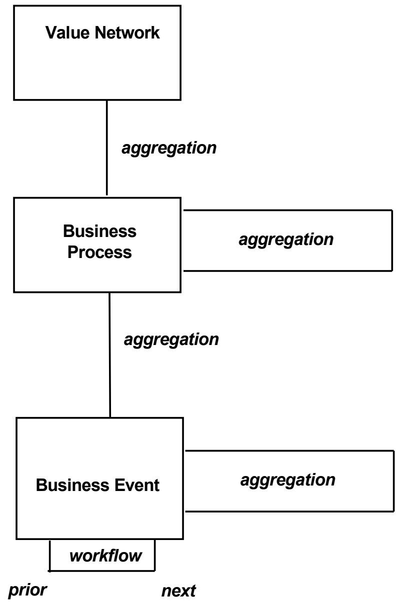


Figure 6-25 – REA Granularity Levels for Independent View

As an overview of where we think REA has been and where it is headed vis-à-vis the capabilities of the double-entry accounting model, we present (somewhat speculatively) Figure 6-26 which is based on the path-dependent change and path-creation ideas discussed by Garud, Kumaraswamy, & Karnøe (2010).

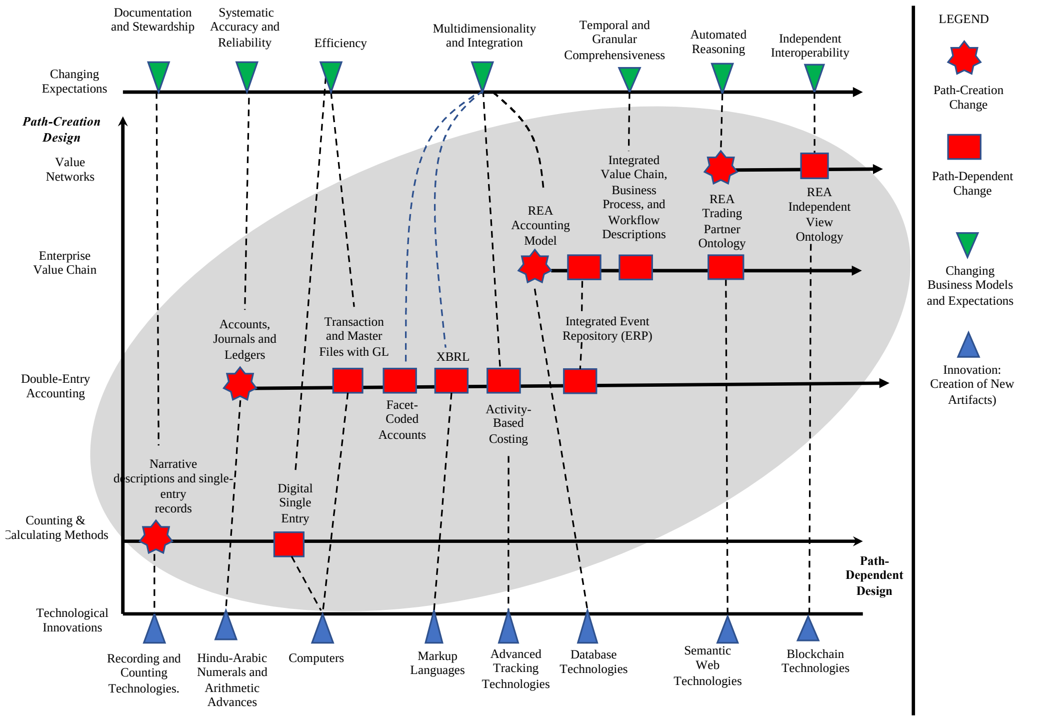


Figure 6-26 – Evolution of Accounting Systems in Terms of Path-Creating and Path-Dependent Design

- On the left vertical axis, we show four different design frameworks as paths for accounting system change. The initial **burst** icon represents path creation while **rectangle** icons represent path dependent change or conceptual lock-in.
- On the lower horizontal axis, we show (with upward-facing triangles) technological innovations that support better types of accounting systems, while the top horizontal line illustrates (with downward-facing triangles) business models and expectations which change as commerce evolves and more is demanded of financial systems (some of which quite simply cannot be accommodated by older systems).

Progress proceeds upward and to the right. Not surprisingly, we speculate that the enterprise value chain path and the value network path supported by REA ontological structures will fare far, far better in a business environment of the future dominated by automated reasoning software and block chain trading venues.

As a final message to readers, we note that both the original REA paper and this monograph's preface begin with an enumeration of four important weaknesses of the conventional accounting model (McCarthy, 1982 p. 554-55):

1. Its dimensions are limited. Most accounting measurements are expressed in monetary terms: a practice that precludes maintenance and use of productivity, performance, reliability, and other multidimensional data.
2. Its classification schemes are not always appropriate. The chart of accounts for a particular enterprise represents all of categories into which information concerning economic affairs may be placed. This will often lead to data being left out or classified in a manner that hides its nature from non-accountants.
3. Its aggregation level for stored information is too high. Accounting data is used by a wide variety of decision makers, each needing differing amounts of quantity, aggregation, and focus depending upon their personalities, decision styles, and conceptual structures. Therefore, information concerning economic events and objects should be kept in as elementary a form as possible to be aggregated by the eventual user.
4. Its degree of integration with the other functional areas of an enterprise is too restricted. Information concerning the same set of phenomena will often be maintained separately by accountants and non-accountants, thus leading to inconsistency plus information gaps and overlaps.[65]

We contend that, despite many technological advances, **<u>all</u>** of these criticisms remain true today (although some modicum of relief has been afforded by ERP systems which have come on the scene since 1982). That unmoving assessment of all four negative points has been seconded recently by the IMA in its conceptual costing structures work (White, et al. 2014).

In 2019, we would expand this critical list to include two more major weaknesses of account-based double-entry:

[65] These four weaknesses were initially enumerated in McCarthy (1980, p. 628)

5. It cannot support automated reasoning because its embedded semantics are either non-existent or very weak. This is a key feature of ontologies, so double-entry accounts will probably never advance that far up the interoperability scale (see Figure 6-7).
6. It cannot operate well in an independent-view distributed business transaction repository, because the fundamental accounting equation (A=L + OE), on which all accounts are based, is a trading-partner view of commerce. Taking one trading partner's ledger accounts and balancing them with another trading partner's ledger simply does not work easily, and it may involve massive amounts of rework to achieve an accounting balancing act that is meaningless in open collaboration space.

As we contend in the top right of Figure 6-26, the REA ontology presented here handles these last two critical features with great facility, in addition to overcoming the original four criticisms as well. We hope that monograph readers use those strengths as a motivation to study and learn all of the features we have presented in these six chapters, and further, to investigate research possibilities for accounting in this new commercial arena.

References

Abrial, J.-R. (1974). Data Semantics. In J. Klimbie, & K. Koffeman (Eds.), *Data Base Management* (pp. 1-60). Amsterdam: North Holland.

Armitage, H. (1985). *Linking Management Accounting Systems with Computer Technology.* Hamilton, Ontario: The Society of Management Accountants of Canada.

Benkler, Y. (2002). Coase's Penguin, or Linux and "The Nature of the Firm". *The Yale Law Journal, 112*(3), 369-446.

Bennett, M. (2015). Defining Equity and Debt Using REA Claim Semantics. *Value Modeling and Business Ontologies Workshop.* Vienna, AT.

Bennett, M., Gilmore, M., & Nehmer, R. (2018). *Claims, Debt, and Equity in REA and FIBO: Proposed Extensions.* Bilbao, ES: Proceedings of the Semantic Shed Workshop.

Bergholtz, M., Jayaweera, P., Johannesson, P., & Wohed, P. (2004). Bringing Speech Acts into UMM. *1st International REA Technology Workshop.* Copehhagen, DK: Stockholm University and Royal Institute of Technology.

Black, J. D., & Black, A. G. (1929). *Production Organization.* New York: H. Holt and Company.

Boubaker, A., Leshob, A., Mili, H., & Charif, Y. (2017). A Pattern Based Approach to Extract REA Value Models from Business Process Models. *Intelligent Systems in Accounting, Finance, and Management, 24*(1), 29-48.

Brodie, M. L., Mylopoulos, J., & Schmidt, J. W. (Eds.). (1984). *On Conceptual Modeling.* New York: Springer-Verlag.

Brown, R. G. (2018, August). *A Simple Model for Smart Contracts*. Retrieved from Richard G. Brown: https://gendal.me/2015/02/10/a-simple-model-for-smart-contracts/

Clancey, W. J. (1985). Heuristic Classification. *Artificial Intelligence*, *27*(3), 289-350.

Coad, P. (1995). *Object Models: Strategies, Patterns, and Applications.* Englewood Cliffs, NJ : Prentice-Hall.

Coase, R. H. (1937). The Nature of the Firm. *Economica*, *4*(16), 386-405.

Codd, E. F. (1970). The Relational Model of Data for Large Shared Data Banks. *Communications of the ACM, 13*(6), 377-387.

Committee of Sponsoring Organizations of the Treadway Commission. (1992). *Internal Control - Integrated Framework.* Jersey City, NJ: American Institute of Certified Public Accountants.

Committee of Sponsoring Organizations of the Treadway Commission. (2004). *Enterprise Risk Management - Integrated Framework.* Durham, NC: American Institute of Certified Public Accountants.

Committee of Sponsoring Organizations of the Treadway Commission. (2009). *Internal Control - Integrated Framework Guidance on Monitoring Internal Control Systems Volume II: Application.* Durham, NC: AICPA.

Committee of Sponsoring Organizations of the Treadway Commission. (2013). *Internal Control - Integrated Framework.* Durham, NC: American Institute of Certified Public Accountants.

Committee of Sponsoring Organizations of the Treadway Commission. (2016). *Enterprise Risk Management: Aligning Risk with Strategy and Performance.* NY: COSO.

Committee of Sponsoring Organizations of the Treadway Commissions. (2008). *Guidance on Monitoring Internal Control Systems Volume II - Guidance.* Durham, NC: American Institute of Certified Public Accountants.

Davenport, T. (1993). *Process Innovations; Reengineering Work Through Information Technology*. Boston: Harvard Business School Press.

David, J. S. (1997). Three 'Events' That Define a REA Methodology for Systems Analysis, Design, and Implementation. *Proceedings of the Annual Meeting of the American Accounting Association.* American Accounting Association.

David, J. S., & McCarthy, W. E. (1992). *Ventura Vehicles Case Description.* East Lansing: Michigan State University.

David, J. S., Gerard, G., & McCarthy, W. E. (2002). Design Science: An REA Perspective on the Future of AIS. In S. Sutton, & V. Arnold (Eds.), *Researching Accounting as an Information Systems Discipline* (pp. 35-63). Sarasota: American Accounting Association.

David, J. S., McCarthy, W. E., & Sommer, B. S. (2003). Agility - The Key to Survival of the Fittest in the Software Market. *Communications of the ACM, 46*(5), 65-69.

Denna, E. L., Jasperson, J., Fong, K., & Middleman, D. (1994). Modeling Conversion Process Events. *Journal of Information Systems, 8*(1), 43-54.

Denna, E., & L. McCarthy, W. E. (1987). An Events Accounting Foundation for DSS Implementation. In C. W. Holsapple, & A. B. Whinston (Eds.), *Decision Support Systems: Theory and Application* (pp. 239-263). Berlin, DE: Springer-Verlag.

Denna, E., Cherrington, J., Andros, D., & Hollander, A. S. (1993). *Event-Driven Business Solutions:Today's Revolution in Business and Information Technology.* Homewood, IL: Irwin.

Dumas, D., & Alexander, P. A. (2016). Calibration of the Test of Relational Reasoning. *Psychological Assessment, 28*(10), 1202-1318.

Dunn, C. (2012). *Resources-Events-Agents: An Ontology for Designing, Controlling, and Using Integrated Enterprise Systems.* McGraw Hill Create.

Dunn, C. L., & McCarthy, W. E. (1992). Conceptual Models of Economic Exchange Phenomena: History's Third Wave of Accounting Systems. *Collected Papers of the Sixth World Congress of Accounting Historians*, (pp. 133-164). Kyoto, JP.

Dunn, C. L., & McCarthy, W. E. (1997). The REA Accounting Model: Intellectual Heritage and Prospects for Progress. *Journal of Information Systems, 11*, 31-51.

Dunn, C. L., Gerard, G. J., & Grabski, S. V. (2016). Resources-Events-Agents Design Theory: A Revolutionary Approach to Enterprise System Design". *Communications of the Association for Information Systems, 38*.

Fedorowicz, J., & McCarthy, W. E. (1983). Derivation of Cost Accounting Data in a Relational Accounting System. *American Institute of Decision Sciences.* San Antonio: American Institute of Decision Sciences.

Ferrucci, D., Brown, E., Chu-Caroll, J., Fan, J., Gondek, D., Kalyanpur, A. A., . . . Welty, C. (2010). Building Watson: An Overview of the DeepQA Project. *AI Magazine, 31*(3), 59-79.

Fikes, R., & Kehler, T. (1985). The Role of Frame-based Representation in Reasoning. *Communications of the ACM, 28*(9), 904-920.

Fillmore, C. J. (1967). The Case for Case. In E. Bach, & R. Harms (Eds.), *Proceedings of the Texas Symposium on Language Universals* (p. 135). New York: Holt, Rinehart and Winston.

Financial Accounting Standards Board. (1985). *Statement of Financial Accounting Concepts No. 6 - Elements of Financial Statements.* Norwalk, CT: Financial Accounting Standards Board.

Financial Accounting Standards Board. (2010). *Statement of Financial Accounting Concepts No. 8 - Conceptual Framework for Financial Reporting.* Norwalk, CT: Financial Accounting Standards Board.

Fischer-Pauzenberger, C., Ondra, M., & Schwaiger, W. (2018). OntoREA© Accounting and Financial Model: Including Option Contracts in the MDD-Context. *12th International Workshop on Value Modeling and Business Ontologies.* Amsterdam, NL.

Fisher, I. (1906). *The Nature of Capital and Income.* New York: The MacMillian Co.

Foster, L. (2017). *NRP Round Trip.* Retrieved from http://mikorizal.org: https://speakerdeck.com/mikorizal/nrp-round-trip

Fowler, M. (1997). *Analysis Patterns; Reusable Object Models.* Menlo Park: Addison Wesley.

Gal, G. (2015). Data Modeling and the Audit. *International Journal of Auditing Technology, 2*(4), 316-332.

Gal, G., & McCarthy, W. E. (1980). A CODASYL Implementation of an Events Accounting System. *Midwest American Accounting Association Meeting.* Chicago: American Accounting Association.

Gal, G., & McCarthy, W. E. (1982a). Operation of a Relational Accounting System. *Northeast American Acccounting Association Meeting.* Newport: American Accounting Association.

Gal, G., & McCarthy, W. E. (1982b). Specification of Internal Controls with QBE Views and Dictionary Facilities. *Midwest Meeting of the American Institute for Decision Sciences.* Milwaukee: American Institute of Decision Sciences.

Gal, G., & McCarthy, W. E. (1983). Procedural and Declarative Features of a CODASYL Database. In P. Chen (Ed.), *Entity-Relationship Approach to Information Modeling and Analysis* (pp. 197-214). Saugus, CA: ER Institute.

Gal, G., & McCarthy, W. E. (1985a). An Artificial Intelligence Framework for Conceptual Specification and Evaluation of Internal Control. *National TIMS-ORSA Meeting.* Boston.

Gal, G., & McCarthy, W. E. (1985b). Specification of Internal Accounting Controls in a Database Environment. *Computers and Security*, 23-32.

Gal, G., & McCarthy, W. E. (1986). Operation of a Relational Accounting System. *Advances in Accounting*, 83-112.

Gal, G., & McCarthy, W. E. (1991). Semantic Specification and Automated Enforcement of Internal Control Procedures within Accounting Systems. *European Accounting Congress.* Maastricht, NL: European Accounting Association.

Gal, G., Geerts, G., & McCarthy, W. E. (2009). Semantic Specification and Automated Enforcement of Internal Control Procedures within Accounting Systems. *Valuation Management and Business Ontology Workshop.* 2009.

Gamallo, P. (2013). Lexical Inheritance with Meronymic Relationships. *Axiomathes: An International Journal in Ontology and Cognitive Systems, 23*(1), 165-185.

Garud, R., Kumaraswamy, A., & Karnøe, P. (2010). Path Dependence or Path Creation. *Journal of Management Studies, 47*(4), 761-774.

Geerts, G. L. (1993). *Toward a New Paradigm in Structuring and Processing Accounting Data.* Brussels, BE: Free University of Brussels.

Geerts, G. L., & McCarthy, W. E. (1992a). The Cost Revolution from a Data Modeling Point of View. *The Fifteenth Annual Congress of the European Accounting Association.* Madrid.

Geerts, G. L., & McCarthy, W. E. (1992b). Database Accounting Support for Decision Problems in the New Manufacturing Environment. *The International Conference on Economics/Management and Information Technology.* Tokyo, JP.

Geerts, G. L., & McCarthy, W. E. (1994). The Economic and Strategic Structure of REA Accounting Systems. *300th Anniversary Program, Martin Luther University.* Halle-Wittenberg.

Geerts, G. L., & McCarthy, W. E. (1997a). Accounting as Romance: Patterns of Unrequited Love and Incomplete Exchanges in Life and in Business Software. *OOPSLA.* Atlanta: ACM.
Geerts, G. L., & McCarthy, W. E. (1997b). Modeling Business Enterprises As Value-Added Process Hierarchies. In J. V. Sutherland, C. Casanave, J. Miller, P. Patel, & G. Hollowell (Ed.), *Business Object Design and Implementation* (pp. 94-113). Berlin: Springer.
Geerts, G. L., & McCarthy, W. E. (1999). An Accounting Object Infrastructure for Knowledge-Based Enterprise Models. *IEEE Intelligent Systems, July/August*, 89-94.
Geerts, G. L., & McCarthy, W. E. (2000a). Augmented Intensional Reasoning in Knowledge-Based Accounting Systems. *Journal of Information Systems, 14*(2), 127-150.
Geerts, G. L., & McCarthy, W. E. (2000b). *The Ontological Foundation of REA Enterprise Information Systems.* Retrieved from http://www.mus.edu/user/mccarth4
Geerts, G. L., & McCarthy, W. E. (2001). Using Object Templates from the REA Accounting Model to Engineer Business Processes and Tasks. *The Review of Business Information Systems, 5*(4), 89-108.
Geerts, G. L., & McCarthy, W. E. (2002). An Ontological Analysis of the Primitives of the Extended-REA Enterprise Information Architecture. *International Journal of Accounting Information Systems, 3*(1), 1-16.
Geerts, G. L., Graham, L. E., Mauldin, E. G., McCarthy, W. E., & Richardson, V. (2013). Integrating Information Technology into Accounting Research and Practice. *Accounting Horizons, 27*(4), 815-840.
Geerts, G., & McCarthy, W. E. (1992c). The Extended Use of Intensional Reasoning and Epistemologically Adequate Representations in Knowledge-based Accounting Systems. *Proceedings of the Twelfth International Workshop on Expert Systems and Their Applications,* (pp. 321-332). Avignon, FR.
Geerts, G., & McCarthy, W. E. (2006). Policy-level Specification in REA Enterprise Information Systems. *The Journal of Information Systems, 20*(2), 37-63.
Geijsbeek, J. B. (1914). *Ancient Double-Entry Bookkeeping.* Houston: Scholars Book Club.
Goldstein, R. C., & Storey, V. C. (1994). Materialization. *IEEE Transactions on Knowledge and Data Engineering, 6*(5), 835-842.
Gordijn, J. (2002). *Value-based Requirements Engineering: Exploring Innovative e-commerce Ideas.* Amsterdam, NL: Vrije Universiteit.
Grabski, S. V., & Marsh, R. J. (1994). Integrating Accounting and Manufacturing Information Systems: An ABC and REA-Based Approach. *Journal of Information Systems, 8*(2), 61- 80.
Gruber, T. R. (1993). A Translatable Approach to Portable Ontologies. *Knowledge Acquisition, 5*(2), 199-220.
Hammer, M. (1990, July/August). Reengineering Work: Don't Automate, Obliterate. *Harvard Business Review, 68*(4), 104-112.
Hammer, M., & Champy, J. (1993). *Reengineering the Corporation.* New York: Harper Business.
Haugen, R. (2002). The Commitment Oriented Orchestration Layer -- Managing Your Business Commitments. *UN/CEFACT Quarterly Meeting.* Seattle, WA.
Haugen, R. (2014). *Value Network Equations.* Retrieved from P2P Foundation: http://wiki.p2pf.org/Value_Network_Equation
Haugen, R., & McCarthy, W. E. (2001a). The Supply Chain Modeling Methodology. *The Automobile Industry Action Group Modeling Workshop.* Southfield, MI: AIAG.
Haugen, R., & McCarthy, W. E. (2001b). REA: Semantic Model for Internet Supply Chain Collaboration. *OOPSLA.* Minneapolis, MN: ACM.

Hay, D. C. (1996). *Data Model Patterns.* New York: Dorsett House Publishing.
Hergert, M., & Morris, D. (1989). Accounting Data for Value Chain Analysis. *Strategic Management Journal*, *10*(2), 175-188.
Hevner, A., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. *MIS Quarterly, 28*(1), 75-105.
Horiuchi, S., & McCarthy, W. E. (2011). An Ontology-based State Machine for Catalog Orders. *Value Modeling and Business Ontologies Workshop.* Ghent, BE.
Horiuchi, S., & Shimizu, X. (2016). Rethinking the Development of a Business Process State Machine. *Shogaku Ronsan*, 593-657.
Horngren, C. T., Datar, S. M., & Rajan, M. V. (2015). *Cost Accounting: A Managerial Approach.* Upper Saddle River, NJ: Pearson Education.
Hruby, P. (2007). *Model-Driven Design Using Business Patterns.* New York; Berlin: Springer-Verlag.
Hruby, P., & Kiehn, J. (2006). Revised Classification of the Enterprise Information Architecture Elements. *3rd International Conference on Enterprise Systems and Accounting.* Santorini Island, GR.
Hunka, F., & Zacek, J. (2015). A New View of REA State Machine. *Applied Ontology, 10*(1), 25-39.
Ijiri, Y. (1975). *Theory of Accounting Measurement.* Sarasota, FL : American Accounting Association.
International Standards Organization. (2007). *ISO/IEC 15944-4 - Information Technology - Business Operational View - Part 4: Business Transaction Scenarios - Accounting and Economic Ontology*. Geneva, CH: International Standards Organization.
International Standards Organization. (2010). *ISO/IEC 14662 Information Technology -- Open-edi Reference Model.* Geneva, CH: International Standards Organization.
International Standards Organization. (2015). *ISO/IEC 15944-4 - Information Technology - Business Operational View - Part 4: Business Transaction Scenarios - Accounting and Economic Ontology: Second Edition.* Geneva, CH: International Standards Organization.
International Standards Organizations. (2001). *ISO/IEC15944-1 - Information Technology - Business Operational View - Part 1: Operational Aspects of Open-edi for Implementation.* Geneva, CH: International Standards Organizations.
International Standards Organizations. (2019). *ISO/IEC 15944-15 Information Technology - Business Operational View - Part 15: Application of Open-edi Business Transaction Ontology in Distributed Business Transaction Repositories and Open Value Networks (COMMITTEE SPECIFICATION WORKING DRAFT).* Geneva, CH: International Standards Organizations.
ISACA . (2012). *COBIT 5 - Enabiling Processes.* Rolling Meadows, IL USA: ISACA.
ISACA. (2019). *COBIT 2019 Framework: Governance and Management Objectives.* Schaumburg, IL: ISACA.
IT Governance Institute. (2007). *COBIT 4.1.* Rolling Meadows, IL USA: IT Governance Institute.
Ito, S., & Vymětal, D. (2013). The Formal REA Model at the Operational Level. *Applied Ontology*, *8*(4), 275-300.
Keynes, J. M. (1936). *The General Theory of Employment, Interest, and Money.* London: Macmillan and co., limited.
Kroszner, R. S., & Putterman, L. (2009). *The Economic Nature of the Firm* (3rd ed.). Cambridge: Cambridge University Press.
Lamport, L. (1978). Time, Clocks, and the Ordering of Events in Distributed Systems. *Communications of the ACM, 21*(7), 558-565.

Lancaster, K. J. (1966). A New Approach to Consumer Theory. *Journal of Political Economy, 74*(2), 132-157.
Lancaster, K. J. (1975). Socially Optimal Product Differentiation. *American Economic Review, 65*(9), 567-585.
Laurier, W., Kiehn, J., & Polovina, S. (2018). REA2: A Unified Formalisation of the Resource-Event-Agent Ontology. *Applied Ontology, 13*(3), 201-224.
Luschei, E. C. (1962). *The Logical Systems of Lesniewski.* Amsterdam: North-Holland.
Mattessich, R. (1964). *Accounting and Analytical Methods.* Homewood, IL: Richard D. Irwin.
McCarthy, J., & Hayes, P. J. (1969). Some Philosophical Problems from the Standpoint of Artificial Intelligence. In D. Michie, & B. Meltzer (Eds.), *Machne Learning* (Vol. 4, pp. 463-502). New York: Wiley.
McCarthy, W. E. (1979). An Entity-Relationship View of Accounting Models. *The Accounting Review, 54*(4), 667-686.
McCarthy, W. E. (1980). Construction and Use of Integrated Accounting Systems with Entity-Relationship Modeling. In P. P. Chen (Ed.), *Entity-Relationship Approach to Systems Analysis and Design* (pp. 625-637). Amsterdam: North-Holland.
McCarthy, W. E. (1982). The REA Accounting Model: A Generalized Framework for Accounting Systems in a Shared Data Environment. *The Accounting Review, 57*(3), 554-578.
McCarthy, W. E. (1984, April). Materialization of Account Balances in the REA Accounting Model. *Invited Presentation to the British Accounting Association*. Norwich, GB.
McCarthy, W. E. (1987). *On the Future of Knowledge-Based Accounting Systems.* University of Missouri, The D.R. Scott Memorial Lecture Series.
McCarthy, W. E. (1999). The NICOSYS Teaching Case. *Semantic Modeling of Accounting Phenomena Workshop.* San Diego.
McCarthy, W. E. (2000). REA Accounting Models -- The Need for Both Internal and External Views in a World of Interconnected Enterprises. *Keynote presentation to the Third European Conference on Accounting Information Systems.* Munich.
McCarthy, W. E. (2003a, November). The REA Modeling Approach to Teaching Accounting Information Systems. *Issues in Accounting Education, 18*(4), 427-441.
McCarthy, W. E. (2003b). The REA Accounting and Economic Ontology: Proposed State Machine. *First UBAC (Uniform Business Agreements and Contracts) Workshop.* Stockholm, Sweden.
McCarthy, W. E. (2003c). On the Partial Mapping of Mediated REA Scenarios in E-Commerce Collaboration Space. *The University of Kansas XBRL Symposium.* Lawrence, Kansas.
McCarthy, W. E. (2005). Western Michigan Office Furniture Teaching Case. *Semantic Modeling of Accounting Phenomena Workshop.*
McCarthy, W. E. (2006). Nantasket Casket, Basket, and Gasket Teaching Case. *Semantic Modeling of Accounting Phenomena Workshop.* Phoenix.
McCarthy, W. E. (2012). Accounting Craftspeople vs. Accounting Seers: Exploring the Relevance and Innovation Gaps in Academic Accounting Research. *Accounting Horizons, 26*(4), 833-843.
McCarthy, W. E., & Outslay, E. (1989). An Analysis of the Applicability of Artificial Intelligence Techniques to Problem Solving in Taxation Domains. *Accounting Horizons, 3*(2), 14-27.
McCarthy, W. E., & Rockwell, S. (1989). The Integrated Use Of First-Order Theories, Reconstructive Expertise, And Implementation Heuristics In An Accounting

Information System Design Tool. *Proceedings of the Ninth International Avignon Workshop on Expert Systems and Their Applications*, (pp. 537-548). Avignon, FR.
Minsky, M. (1975). A Framework for Representing Knowledge. In P. Winston (Ed.), *The Psychology of Computer Vision.* (pp. 211–277). New York: McGraw Hill.
Motschnig-Pitrik, R. (1993). The Semantics of Parts Versus Aggregates in Data/Knowledge Modelling. *CAiSE '93 Proceedings of Advanced Information Systems Engineering Conference* (pp. 352-73). London, UK: Springer-Verlag .
Nakamoto, S. (2008). *Bitcoin: A Peer-to-Peer Electronic Cash System.* Retrieved from Bitcoin: https://bitcoin.org/bitcoin.pdf
Nittler, M. (2018). A Cloud Journey. *3rd Journal of Information Systems Research Conference.* Raleigh: American Accounting Associaton.
Object Management Group. (2011). *Business Process Modeling Notation Version 2.0.* Needham, MA: Object Managment Group.
Object Management Group. (2014). *Interaction Flow Modeling Language.* Needham, MA: Object Management Group.
Object Managment Group. (2012). *Unified Modeling Language.* Needham, MA: Object Management Group.
Obrst, L. (2003). Ontologies for Semantically Interoperable Systems. *Conference on Information and Knowledge Management* (pp. 366-369). New Orleans, LA USA: ACM.
Odell, J. J. (1998). *Advanced Object-oriented Analysis and Design Using UML.* Cambridge: Cambridge University Press.
Ontolog Forum. (2011). *Ontology Summit 2011 Communique: Making the Case for Ontology*. Ontolog Forum.
Organization for the Advancement of Structured Infomation Standards. (2013, November 4). *Universal Business Language Version 2.1.* Retrieved from OASIS: http://docs.oasis-open.org/ubl/UBL-2.1.html
Organization for the Advancement of Structured Information Standards. (2006). *ebXML - Enabiling a Global Market Place*. Retrieved December 9, 2015, from http://www.ebxml.org/geninfo.htm
Porter, M. (1985). *Competitive Advantage: Creating and Sustaining Superior Performance.* New York: Free Press.
Public Company Accounting Oversight Board. (2007). *Auditing Standard No. 5 - An Audit of Internal Control over Financial Reporting That is Integrated with An Audit of Financial Statements.* Washington, DC: Public Company Accounting Oversight Board.
PWC. (2017). *Distinguishing Liabilities from Equity.* New York: PWC.
Quine, W. v. (1969). *Ontological Relativity and Other Essays.* New York: Columbia University Press.
Quine, W. v. (1992). *Pursuit of Truth (revised edition).* Cambridge, MA: Harvard University Press.
Rittenberg, L. E., & Schwieger, B. (2001). *Auditing Concepts for a Changing Environment* (Third ed.). New York: Harcourt, Inc.
Rockwell, S. R., & McCarthy, W. E. (1999). REACH: Automated Database Design Integrating First-Order Theories, Reconstructive Expertise, and Implementation Heuristics for Accounting Information Systems. *International Journal of Intelligent Systems in Accounting, Finance, and Management, 8*(3), 181-197.
Rzhetsky, A., & Evans, J. A. (2011). War of Ontology Worlds: Mathematics, Computer Code, or Esperanto? *PLOS Computational Biology*, *7*(9), 1-4.

Sakai, H. (1981). A Method for Defining Information Structures and Transactions in Conceptual Schema Design. *Proceedings of the Seventh International Conference on Very Large Data Bases* (pp. 225-234). Cannes, FR: VLDB Endowment.

Sensorica. (2019, February). *Sensorica.* Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Sensorica

Simon, H. A. (1969). *The Sciences of the Artificial.* Cambridge, MA: MIT press.

Smith, D., & Smith, J. M. (1977a). Database Abstractions: Aggregation and Generalization. *ACM Transactions on Database Systems,* 105-133.

Smith, J. M. (1978). A Normal Form for Abstract Syntax. *Proceedings of the Fourth International Conference on Very Large Data Bases* (pp. 156-162). West Berlin: IEEE Computer Society.

Smith, J. M., & Smith, D. C. (1977b). Database Abstractions: Aggregations. *Communications of the ACM, 20*(6), 405-413.

Sowa, J. F. (1984). *Conceptual Structures. Information Processing in Mind and Machine.* Reading, MA: Addison-Wesley.

Sowa, J. F. (2000). *Knowledge Representation: Logical, Pholosophical, and Computational Foundations.* Pacific Grove, CA: Brooks/Cole.

Storey, V. C. (1993). Understanding Semantic Relationships. *The VLDB Journal — The International Journal on Very Large Data Bases, 2*(4), 455-488.

Sugumaran, V., & Storey, V. C. (2002). Ontologies for Conceptual Modeling: Their Creation, Use, and Managment. *Data and Knowledge Engineering, 42*(3), 251-271.

Tapscott, D., & Tapscott, A. (2016). *Blockchain Revolution : How the Technology Behind Bitcoin is Changing Money, Business, and the World.* New York: Portfolio/Penguin.

Tarnas, R. (1991). *The Passion of the Western Mind: Understanding the Ideas that have Shaped our World View.* New York: Harmony Books.

The Economist. (2015, October 31). The Great Chain of Being Sure About Things; Blockchains. *The Economist, 417*(8962), pp. 21-24.

The Institute of Management Accountants. (2005). *Conceptual Framework for Mangerial Costing.* Montvale, NJ : The Institute of Management Accountants.

The Institute of Management Accountants. (2012). *Conceptual Framework for Managerial Costing: Draft Report of the IMA Managerial Costing Conceptual Framework Task Force.* Montvale, NJ: Institute of Management Accountants.

Tsichritzis, D. C., & Klug, A. (1978). The ANSISPARC/X3/ DBMS Framework: Report of the Study Group on Database Management Systems. *Information Systems*, 173-91.

Tsichritzis, D. C., & Lochovsky, F. H. (1982). *Data Models.* Englewood Cliffs, NJ: Prentice-Hall, Inc.

United Nations Centre for Trade Facilitation and Electronic Business. (2009). *Core Components Technical Specification Version 3.0.* New York: UN/CEFACT.

United States Congress. (2002). *Sarbanes-Oxley Act.* Washington, DC: United States Congress.

ValueFlows Project. (2017). *Value Flows Principles.* Retrieved from Value Flows Project: https://www.valueflo.ws/introduction/principles.html

Walsh, A. E. (2002). *ebXML: The Technical Reports* . Prentice Hall.

Weber, R. (1986). Data Models Research in Accounting: An Evaluation of Wholesale Distribution Software. *Accounting Review, 61*(3), 498-518.

White, L. R., & Clinton, B. D. (2014). *Conceptual Framework for Managerial Costing.* Montvale, NJ: Institute of Management Accountants.

Winston, M. E., Chaffin, R., & Herrmann, D. (1987). A Taxonomy of Part-Whole Relations. *Cognitive Science, 11*(4), 417-444.

Yu, S. C. (1976). *The Structure of Accounting Theory*. Gainesville, FL: University Presses of Florida.

Zhou, H., Benton, J., Schilling, D. A., & Milligan, G. W. (2011). Supply Chain Integration and the SCOR Model. *Journal of Business Logistics, 32*(4), 332-344.