Enterprise resource planning systems research: The necessity of explicating and examining patterns in symbolic form

A position paper

by

Julia Smith David
Arizona State University
Box 873606
Tempe, AZ 85287-3606
(480) 965-8603 voice
(480) 965-8392 fax
julie.smith.david@asu.edu

Cheryl L. Dunn
351 RBB
Florida State University
Tallahassee, FL 32306-1110
(850) 644-7878 voice
(850) 644-8234 fax
cheryl.dunn@fsu.edu

William E. McCarthy
N270 North Business Complex
Michigan State University
East Lansing, MI 48824-1121
(517) 432-2913 voice
(517) 432-1101 fax
mccarth4@pilot.msu.edu

for Enterprise Management and Resource Planning Systems: Methods, Tools and Architecture 1999: A First International Workshop

Enterprise resource planning systems research: The necessity of explicating and examining patterns in symbolic form

Enterprise resource planning (ERP) systems are certainly being implemented in a growing percentage of companies in recent years, and the study of these systems by MIS and computer science scholars is also gaining prominence. Such systems combine information from various business processes both within and across the separate functional areas of a company to form an integrated enterprise-wide information system.

Researching ERP systems can be done in a normative or a positive fashion with either design science or natural science methodologies (David, Dunn, McCarthy, and Poston 1999). In the case of normative study, this entails the researcher constructing some kind of an ideal model for ERP with a certain set of desirable features, for example: (1) high degree of integration among functional components, (2) minimal redundancy with regard to data modeling and knowledge representation, (3) satisfactory coverage of both traditional transaction processing needs and decision-support-systems-oriented planning needs, or (4) ability to support business process re-engineering initiatives. The researcher might then analyze either specific software components of existing systems or specific implementations within particular companies to produce judgements about how well these packages or implementations measured up to the promulgated criteria. Especially in the case of computer scientists, this type of normative analysis is often followed by construction of working systems (prototypes) which act as proofs of concept or proofs of feasibility to demonstrate the efficacy of new methods, models, constructs, or tools (March and Smith 1995). In the case of positive or descriptive ERP study, the researcher can examine phenomena in the world such as characteristics of ERP systems and productivity of firms and develop theories to explain those phenomena, with the goal of being able to make predictions about future phenomena. Once developed, the theories must be tested as to their predictive ability.

In both the normative and positive cases enumerated above, the primary object of study is either a highly complex piece of software or an implementation of that software in a complicated corporate environment or a combination of both, all of which contain degrees of intricacy and detail that may easily overwhelm even the most technically astute individual or team of researchers. This is evidenced by the vast amount of human, capital, and time resources expended by most companies in an ERP implementation. We argue here that this intricacy and detail cannot be studied well in either a normative or a positive sense without symbolically abstracting from the particular software components or from the particular corporate implementation to some kind of generally agreed upon model of what ERP is or what ERP should be. The fields of enterprise modeling in general (Vernadat 1996; IEMC 1999) and Resource-Event-Agent (REA) enterprise modeling (McCarthy 1982; Geerts and McCarthy 1999) in particular are the symbol sets we propose for this purpose. REA, like much of ERP itself, sprang originally from a highly specific functional environment (multidimensional accounting for the stocks and flows of economic resources), but developed over time into a much more comprehensive control and planning framework. ERP certainly has other functional roots (such as manufacturing resource planning or human resource deployment and planning); the lineage of the underlying model is not as important as the ability of the model to eliminate the "stovepipe" nature of traditional functional areas and to blend the elements of all functional areas together in the most cohesive way possible.

In section 1 of this paper we introduce the Research Pyramid, a framework for research in information systems that was presented by David et al. (1999) and we discuss the implications for studying ERP systems. In section 2 we emphasize the need for ERP studies to explicate and examine patterns in symbolic form (enterprise models) before comparing specific

implementations. In section 3 we defend our choice of the REA accounting model as a valid candidate for an "ideal" symbol set against which ERP symbol sets can be evaluated. In section 4 we discuss ideas for future research.

Section 1: The Investigation of ERP Investment & the Research Pyramid Framework

Do the large-scale investments of time, money, and human resources that managers have made in ERP systems over the course of the last few years make economic sense? Have managers received benefits from ERP that exceed its considerable cost?

To many managers (especially those who have been responsible for these extremely costly projects), the answer to this question is an obvious one. "Of course, they have." They have made the firm more integrated, more agile, more lean, and more modern. However, as noted by scholars like Brynjolfsson and Yang (1999), there have been multiple cases where large computer projects and investments in information technology have not led to demonstrable increases in measures of firm well-offness like profitability and productivity. This does not mean that these investments were not rational and wise; it only means that their efficacy was sometimes hard to demonstrate in small-scale case studies and/or over short periods of time. Brynjolfsson and Yang did in their most recent study (1999) demonstrate that an increase of one dollar in the quantity of computer capital installed by a firm is associated with an increase of on the order of ten dollars in the financial markets' evaluation of the firm. This was certainly a very positive result, but it was done over a long time (eight years) and a large sample (1,000 firms).

A question that occurs to us is whether or not it is possible to study the effects of IT spending on the more focused level of a smaller set of firms, more limited time scales (a single innovation project), and a narrower range of IT innovation and spending (ERP implementation only). We believe that the answer to this question is yes, but we also believe that it is possible to scale down too far and to become too focused on a single firm and technology implementation. To alleviate this danger of becoming too specifically focused, we introduce our idea of the research pyramid next. This is a framework that we introduced specifically for the purpose of widening the focus of information systems study from just the actual companies and their operational information systems to include in an integrated way the notions of abstract symbol sets and mental concepts. Studies like the one noted above worked at a very high level of granularity, and we propose the need to go lower.

The research pyramid proposed by David et al. (1999) is an extension of Sowa's (1999) Meaning Triangle and is depicted in Figure 1. The three outside corners of the pyramid come from Sowa's "Meaning Triangle," illustrating that real-world *objects* (such as those existing in the day-to-day operations of a company called "Sy's Fish") are (1) perceived as *concepts* in the minds of humans and (2) represented as *symbols* in linguistic, paper, or electronic form for communication with other humans. The AIS corner that extends the triangle into a pyramid illustrates that these symbol sets (as representations of perceived objects) can be implemented on computers in modern information systems. This research pyramid can be used to guide information systems research. David et al. (1999) outline various types of research questions that fit along the pyramid's edges, i.e. combinations of the pyramid's primitives such as Object-IS or IS-Symbol. They also discuss the benefit of studying faces of the pyramid, that is, simultaneously studying adjacent edges (e.g. Object-Symbol-IS). ERP research may encompass any of the edges or faces on the research pyramid. Our position is that studying faces of the pyramid that include the Symbol construct will be the most useful for developing theory from which future predictions may be made. Behavioral IS research often involves the IS-Concept edge in isolation, e.g. what effect do systems have on user perceptions, or what effect do system designers' mental models have on the resulting system. Incomplete theory regarding cognitive processes and individual differences often makes it impossible to replicate

results of a study with similar participants and similar systems. Studies encompassing the IS-Symbol-Concept face, rather than just the IS-Concept edge, of the pyramid may be useful for developing human-computer interaction theory. Consistencies in symbol sets may be identified across systems and results of many studies may be compiled to reveal patterns from which theory can be proposed and tested. Similarly, archival, survey, and field study comparisons of the effects of systems on organizations (i.e. the IS-Object) edge are frequently done and are certainly valid research. Unfortunately there are many implementation-specific latent variables and often this type of study yields unexpected results that subsequently cannot be explained. Again, studies encompassing the IS-Symbol-Object face, rather than just the IS-Object edge, may allow identification of symbol set patterns in different systems that can aid in theory development.

Section 2: The Symbolic Basis for ERP Study

As we mentioned in the previous section, it is possible to study the effect of IT spending and innovation as Brynjolfsson and Yang (1999) do (i.e., in the large) using just the "information systems" and "physical reality" components of the pyramid. This works as long as the focus is very broad (i.e., the firm (s) as seen from 10,000 feet). These two components do not suffice however when we research innovations like ERP in the small because of the idiosyncratic and non-random nature of the studied components and their effects. If we could find two identical firms that implemented different ERP solutions under absolutely identical circumstances and produced allegedly different degrees of success, we might be able to make inferences. However far more common is the case where different firms implement different systems under different circumstances with different results. As our pyramid suggests, we think that these different implementations can only be compared at the symbolic level.

The symbolic basis for the way in which we believe ERP should be studied is illustrated in Figure 2 and explained below.

First of all, each ERP system should be symbolically abstracted to produce models of (1) its functional aspects (the things to be done and the way things are done), (2) its information aspects (the way it stores facts and knowledge about the objects of the enterprise, their use and evolution, their links, and their constraints), and (3) its organizational aspects (organizational imperative and decision levels) (Vernadat 1996 chaps. 4-5-7). Without this symbolic abstraction, there is simply no way to make general statements about how one system compares to another, except if one wants to couch these comparisons in terms of general characteristics like system size, system cost, and system installation base. These gross descriptors might be of general interest, but they reveal very little of the mediating purpose of the software designers and very little of the reason why a firm should prefer one system to another.

As illustrated in the figure, the symbolic abstractions of each ERP system can be compared among each other to reveal differences in function, information architecture, and organization. However, as additionally portrayed at the very top of Figure 2, we believe that the greatest benefit in symbolic abstraction would be realized when all of those derived symbol sets are additionally compared against some normative enterprise model. For that norm, we propose an REA information architecture, and we turn next to a brief explanation of that model's components.

Section 3: The REA Model as a Basis for ERP Comparison

REA Definition.

Most ERP researchers come from computer science or management information systems areas and are not familiar with the REA model because of its origins in the accounting

literature. REA certainly holds as a premise that any enterprise-wide information system must be built with accounting phenomena comprising its core; however, the definition of accounting phenomena used by REA researchers is much different than that definition historically used by traditional financial accountants and relegated to accountants by non-accountants (David, Dunn, McCarthy, and Poston 1999). The REA model was proposed by McCarthy (1982) as a framework for the conceptual modeling of accounting phenomena that facilitates the conceptual modeling of enterprise-wide schemata. At that time, REA was a framework at a single level of abstraction. This model has been further developed and today is proposed as not merely a framework for designing systems but as an integral foundation for the daily operations of systems (Geerts and McCarthy 1999). The framework is now developed at three levels of abstraction: the value chain level, the conceptual model level, and the task level. The value chain level is the highest level of abstraction; it provides a description of all of an enterprise's business processes and depicts the integration between the processes via the passing of resources from one process to another.

An example of the value chain level for a company called "Sy's Fish" is portrayed in Figure 3.   Each business process is an oval with at least one input and one output. Each input/output is an instance of the class *economic resource*, and their consumption (shown with a minus sign) and acquisition (shown with a plus sign) are represented by instances of the class *economic event*. Inside and outside parties to the exchanges are represented by instances of the class *economic agents*.   Each process has a patterned representation of **r**esources, **e**vents, and **a**gents; hence the REA name. Figure 3 shows only the event entities for each process.

Workflow elements are represented one more level down from the REA constellations as tasks. For example, the tasks needed to purchase fish (an event) might include some negotiation and some packaging, but the firm shown chooses not to represent them because they appear to be below the level at which management needs to plan, control, and evaluate economic phenomena (Hollander et al. 1999).

REA exchange templates can also be augmented by type and commitment images that increases greatly the percentage of non-transaction data covered by the model's basic components. In the Sy's Fish example of Figure 3, these augmentations might include purchase orders, customer segments, and employee skills.

REA as a Normative ERP Framework
In Figure 2, we speculated that research in the ERP field could be strengthened by taking the symbol sets derived by each ERP package and comparing it to some normative model of what should be included in the representation. Here, we propose REA for that purpose, although it is certainly clear that normative models from other sub-disciplines (such as supply chain management or manufacturing) will need to be considered as well.

The discussion above portrays the ability of the REA model to enable functional integration in a way that is probably unexpected given its origins in the accounting literature. There are no concepts of debits, credits, or accounts in the REA model. Instead it is focused on modeling the integrated business processes of enterprises at varying levels of abstraction. We believe REA is a robust candidate as a completely specified enterprise value chain symbol set to which symbol sets underlying ERP systems may be compared because of its strong semantic and microeconomic heritage.

Figure 4 presents the various functional lineages of ERP software packages. Some ERP systems evolved from traditional accounting systems, changing from coded account systems to file-based general ledger systems and were then integrated with other functional areas to become ERP systems. Others evolved from materials resource planning (MRP) systems to MRPII systems to ERP systems, or from payroll/personnel systems to human resource management systems to ERP systems (David, McCarthy and Sommer 1998). The dots in Figure 4 represent the possibility of ERP systems having roots in still other functional

areas. We believe that the symbol sets underlying ERP systems that evolved from traditional functional areas are less flexible and comprehensive than the REA symbol set for two reasons.

- First, our admittedly limited examination of alternative symbol sets leads us to believe that those developed in other disciplines tend to keep functional areas quite separate from each other, integrating primarily at the reporting level.
- Second, such symbol sets tend to maintain accounting artifacts as base objects rather than deriving them as views from the corresponding business event objects. This is because many business people do not understand that data do not need to be filtered and stored according to the traditional accounting equation Assets = Liabilities + Owners Equity. Although the history of accounting illustrates that accounting was not always accomplished using the double-entry bookkeeping method, the use of double-entry has become entrenched in our society such that many accountants mistake its artifactual constructs as natural phenomena (Dunn and McCarthy 1999).

At the top right of Figure 4, we have illustrated that the integrated set of symbols abstracted from production ERP systems might best be typed and evaluated by comparison to Full-REA norm (Geerts and McCarthy 1992). This is indeed a high standard that exceeds the computational and representation capabilities of all existing software. It assumes that all resources can be represented (even intangible ones), that all entrepreneurial logic is explicit (why do firms support local charities?), and that all exchanges occur at arm's-length (we treat the time of inside employees as if we were purchasing it ourselves in hourly increments). This is a high standard, but using it as an ideal precludes the automatic use of unexamined system compromise. ERP systems can only get better if we keep such lofty ambitions as targets.

Section 4: Future Research Directions

Numerous opportunities exist for innovative ERP research.   However, as we illustrate in our pyramid of Figure 1 and in our abstraction process of Figure 2, we believe that progress in this field will best proceed normatively. Researchers should always set the bar high by demanding additional features suggested by economic analysis but discouraged by implementation possibilities.   Full-REA models provide a good source for this normative analysis.

We also caution researchers in this field to avoid the temptation to casually compare one ERP system against another at the implementation level. This is certainly the level at which software firm and consulting firm marketing aims, but its claims to generalizability are few. Theory development comes from observation of patterns in the world, therefore examination of the patterns in the symbolic form underlying ERP systems will facilitate theory development, and make ERP study more systematic.

One final important direction we see for explicitly-semantic enterprise models of the type we are advocating here is the graduation and expansion of those symbol systems from simple front-end analysis tools that pinpoint the best conceptual schema for an enterprise (eventually to be compromised of course) to their use as active knowledge models that govern the day-to-day operation of the enterprise information system. As illustrated by Guarino (1998), this is the notion of ontology-driven information system operation.
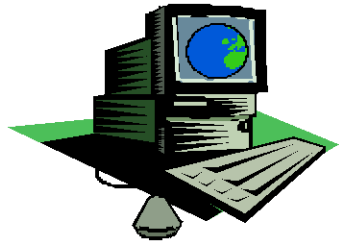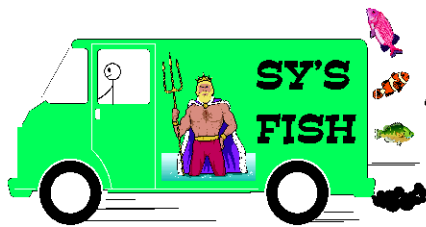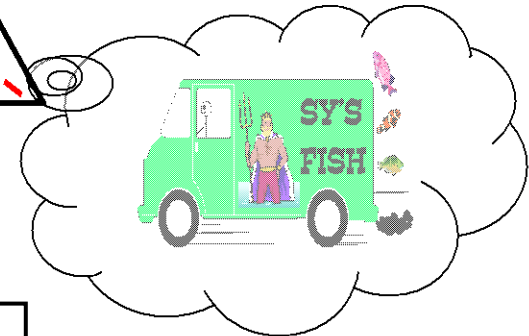
Figure 1 – Research Pyramid
(adapted from David, Dunn, McCarthy, and Poston 1999)
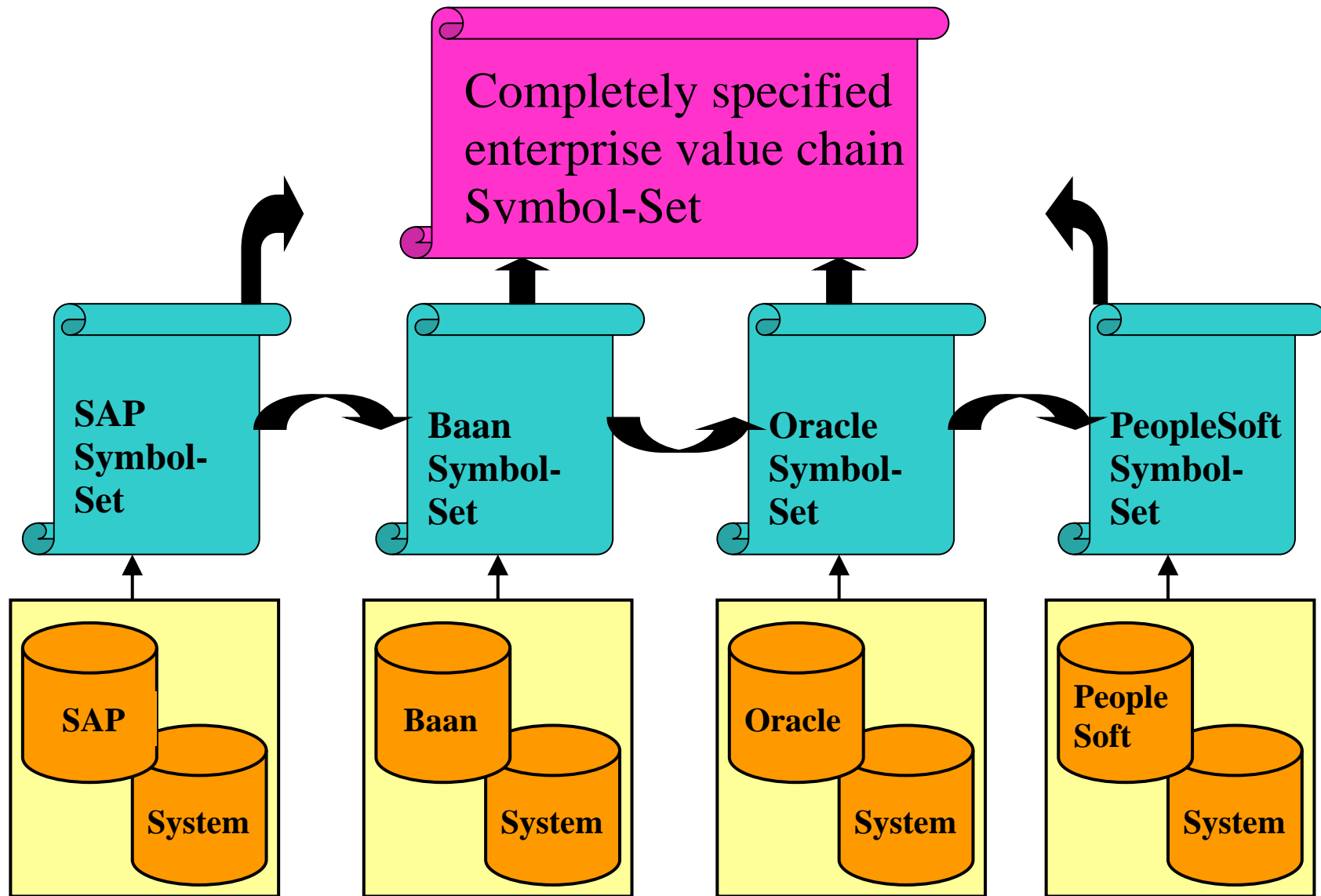
Figure 2 – The Symbolic Basis for ERP Study

Financing Process

+ Cash Receipt — Cash Disbursement **−**

Cash

Cash

Payroll Process

− Cash Dis. — Labor Acquistion **+**

Store Processing

Labor

Buying Process

Cleaning Process

Selling Process

− Cash Dis. — Purchase **+**

Clean — Clean

− Sale — Cash Receipt **+**

Truck Acquistion Process

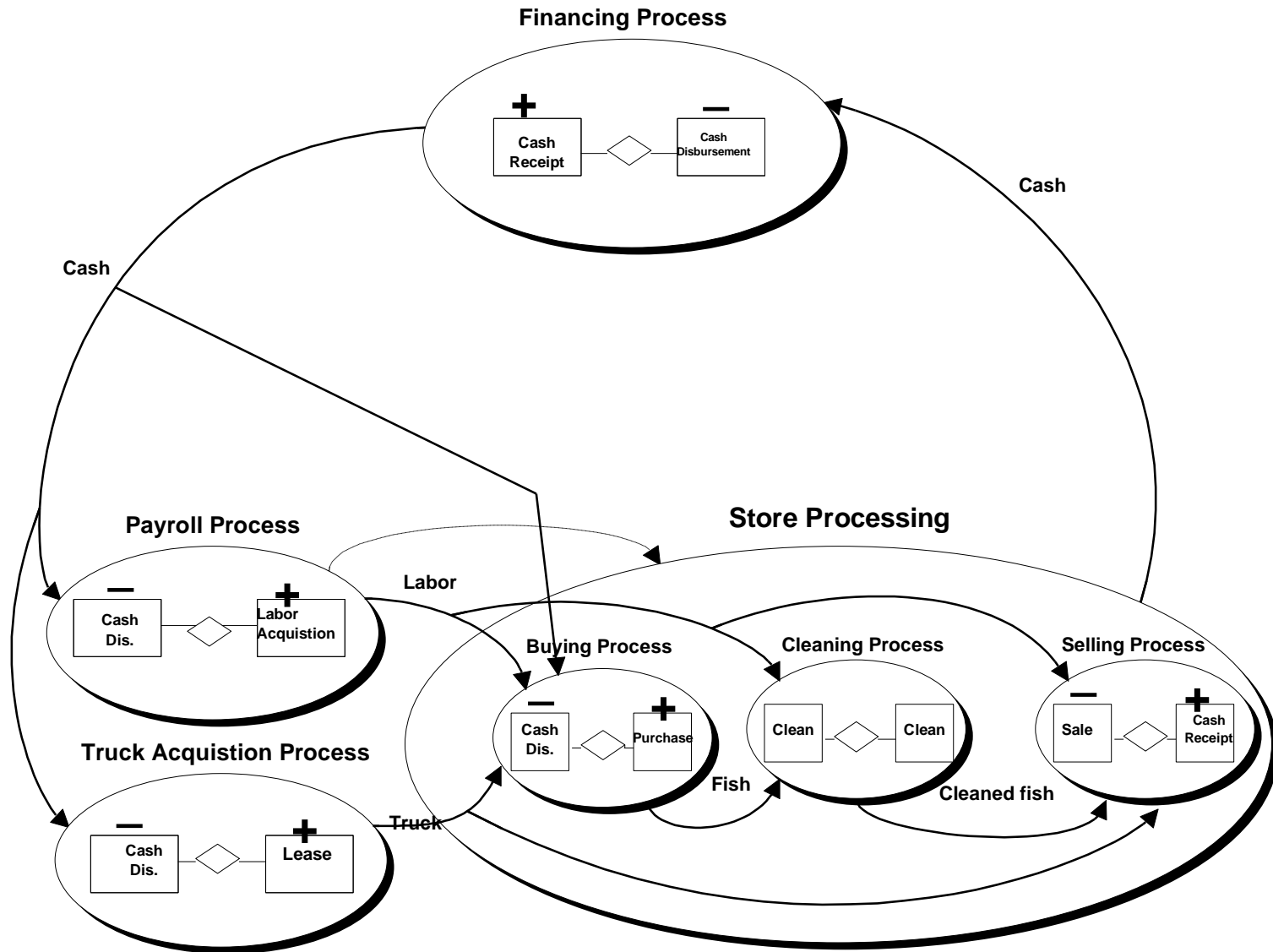**−** Cash Dis. — Lease **+**

Truck

Fish

Cleaned fish

Figure 3 – "Sy's Fish" Value Chain
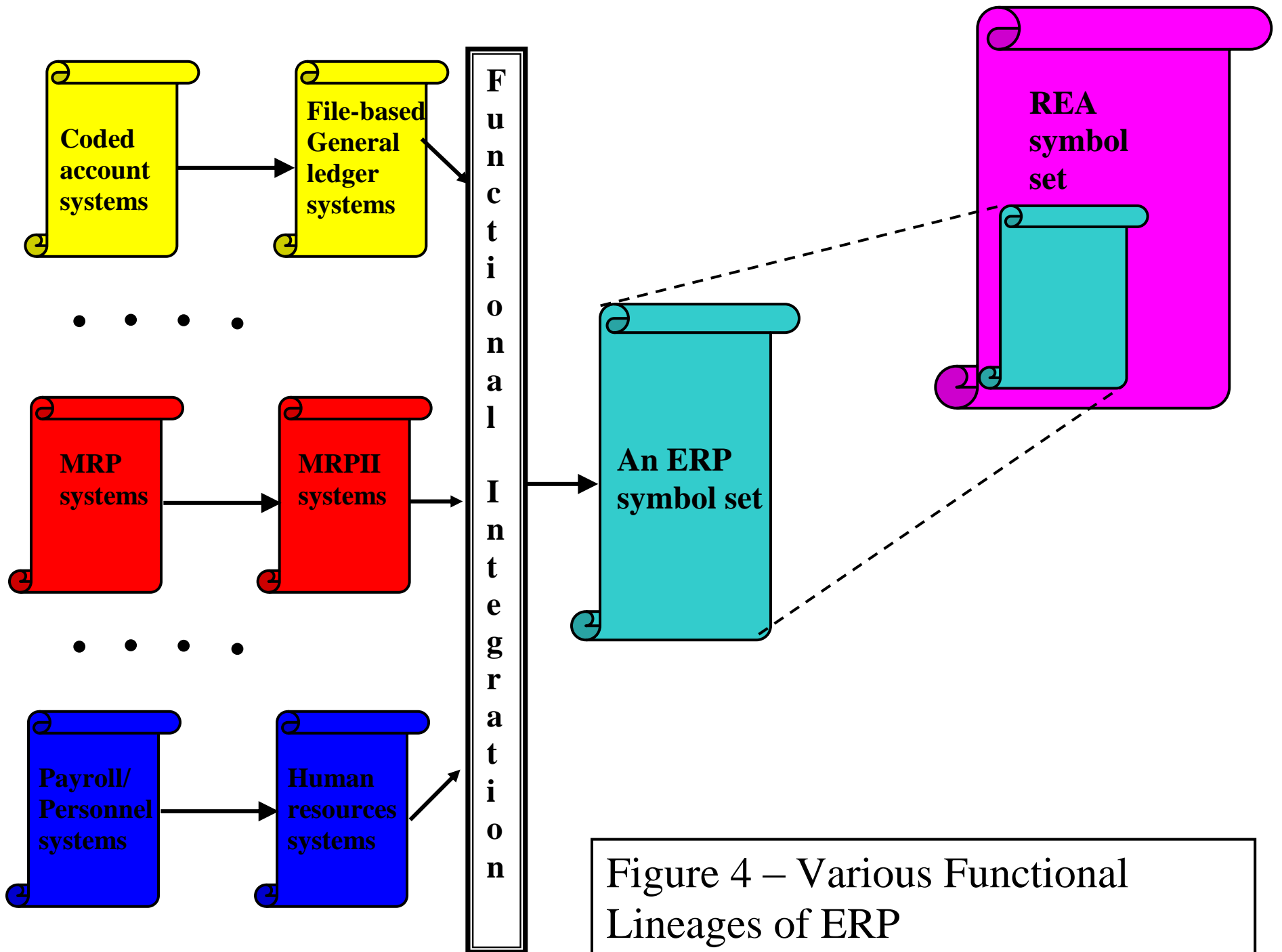From Geerts and McCarthy (1997)

Figure 4 – Various Functional Lineages of ERP

References

Brynjolfsson, E. and S. Yang. 1999. The intangible costs and benefits of computer investments: Evidence from the financial markets. Working paper, MIT Sloan School.

Burnus, P., K. Mertins, and G. Schmidt. Eds. 1998. *Handbook on Architectures of Information Systems*. Springer-Verlag, Berlin∃Heidelberg.

David, J.S., C.L. Dunn, W.E. McCarthy, and R.S. Poston. 1999. The research pyramid: A framework for accounting information systems research. *Journal of Information Systems.* 13 (1) Spring: 7-30.

_____, W.E. McCarthy, and B. Sommer. 1998. The evolution of accounting information systems. Working paper, Arizona State University.

Dunn, C.L. and W.E. McCarthy. 1997. The REA accounting model: Intellectual heritage and prospects for progress. *Journal of Information Systems* (Spring): 31-51.

_____. and _____. 1999. Symbols used for economic storytelling: A progression from artifactual to more natural accounting systems. Working paper, Florida State University.

Geerts, G. and W. E. McCarthy. 1992. 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*, Avignon, France (paper also on 4th submission to the *Journal of Information Systems*).

_____. and _____. 1997. Modeling business enterprises as value-added process hierarchies with Resource-Event-Agent Object templates. *Business Object Design and Implementation*, edited by J. Sutherland and D. Patel, Springer-Verlag, London: 94-113.

_____. and _____. 1999. An accounting object infrastructure for knowledge-based enterprise models. *IEEE Intelligent Systems & Their Applications* (July –August 1999), pp. 1-6.

Guarino, N. 1998. Formal ontology and information systems. *Proceedings of FOIS'98*, Trento, Italy.

Hollander, A.S., E.L. Denna, and J.O. Cherrington. 1999. *Accounting, Information Technology and Business Solutions*, 2nd edition. Richard D. Irwin, Chicago, IL.

IEMC. 1999. *Proceedings of the IFIP International Enterprise Modeling Conference: IEMC'99*. A. Rolstadas, conference chair. Verdal, Norway.

March, S.T. and G.F. Smith. 1995. Design and natural science research on information technology. *Decision Support Systems* 15 (4): 251-267.

McCarthy, W.E. 1982. The REA accounting model: A generalized framework for accounting systems in a shared data environment. *The Accounting Review* (July): 554-578.

Sowa, J. 1999. *Knowledge Representation: Logical, Philosophical, and Computational Foundations.* PWS Publishing.

Vernadat, F.B. 1996. *Enterprise Modeling and Integration: Principles and Applications*. Chapman & Hall. London.