

# A WORK SYSTEM PERSPECTIVE ON ENTERPRISE MODELLING: A THOUGHT EXPERIMENT

*Research in Progress*

Alter, Steven. University of San Francisco, San Francisco, USA, [alter@usfca.edu](mailto:alter@usfca.edu)

## Abstract

*This paper presents a thought experiment related to enterprise modelling (EM). Leaders in the EM community called for “Enterprise Modelling for the Masses” in a position paper at PoEM 2016, the Ninth IFIP 8.1 Working Conference on the Practice of Enterprise Modelling. A revised version appeared in BISE in 2018. The position paper’s vision builds on ambitious assumptions about combining disparate model-related artifacts. That approach articulates interesting research challenges but may prove impractical.*

*This paper presents a different approach to EM for the masses. It shows how an extension of work system theory (WST) and the work system method (WSM) might be used for EM. As research in progress toward a proof of concept paper, it starts by showing how ideas in a metamodel that extends WST provides a potential basis for EM. It outlines a thought experiment in which EM issues in a hypothetical corporate reengineering effort are represented using a modelling language based on a greatly reduced version of the metamodel. The next step is to test variants on the modelling language on other hypothetical examples, and then test whether it is helpful in understanding a real enterprise that is being changed or created.*

*Keywords: Enterprise modelling, Work system theory, Work system metamodel*

## 1 A Step toward “Enterprise Modelling for the Masses”?

This paper was inspired by “Enterprise Modelling for the Masses – From Elitist Discipline to Common Practice,” (Sandkuhl et al. 2016), a position paper presented at the 2016 PoEM conference by widely recognized leaders in the enterprise modelling (EM) community (A revised version appeared in *BISE* as Sandkuhl et al. 2018.) That position paper “argues that EM is far away from reaching its maximum potential.” It says that “EM typically is done by a limited number of people in organizations inclined to methods and modelling. What is captured in models is only a fragment what ought to be captured.” The position paper proposes that “grassroots modelling could lead to groundbreaking innovations in EM.”

EM methods and tools are like lenses in that they focus on and clarify some topics and issues while downplaying or ignoring others. A general challenge for any lens for EM is apparent from the range of topics and issues often associated with enterprise models and enterprise modelling: rigor AND agility AND business architecture AND IT architecture AND information AND systematic and integrated change AND sense making AND communication between stakeholders AND model deployment and activation AND standardization AND documentation. Adding a further challenge, Sandkuhl et al. (2016, p. 228) calls for “lightweight EM approaches that do not necessarily focus on traditional EM qualities like completeness and coherence, but instead on usefulness and impact not only for architects and corporate IT, but also for the majority of organizational stakeholders that might benefit from more professional modelling to support their decentral [sic], focused analysis and design problems.” ... “For modelling to have a larger effect, we propose a move of technologies and approaches to also enable ‘normal’ knowledge workers to be active modelers both by adapting the applications they are using to support their daily work tasks and by providing support for specific nonroutine situations” (p. 229).

**An existing lightweight approach.** Many aspects of Sandkuhl et al. (2016), hereafter called “the position paper,” call for something like an existing lightweight approach whose initial version was described over two decades ago in a paper called “How should business professionals analyze information systems for themselves?” (Alter, 1995). Applications of that approach were reported in a paper (Truex et al., 2010) called “Systems Analysis for Everyone Else: Empowering Business Professionals through a Systems Analysis Method that Fits Their Needs.” That paper describes how 75 working business professionals with extensive business experience used a systems analysis template in MBA assignments that called for analyzing IT-reliant work systems in their own organizations and recommending improvements. In a quite different teaching situation, ideas from the same lightweight approach helped users generate significantly fewer erroneous user stories in a controlled experiment involving 160 Indian undergraduates with little business experience who were learning about agile development (Bolloju et al, 2017). This paper uses the main ideas of a more current description of the initially lightweight approach as the basis of a new perspective on the process and content of EM.

This paper explains that modelling based on work system theory (WST) and the work system method (WSM) is somewhat akin to the type of modelling that the position paper calls for. The entire history of WST/WSM was motivated by similar ambitions, i.e., providing an organized approach that helps business professionals analyse and understand work systems (or enterprises).

**Definition of enterprise modelling.** A definition of EM from Vernadat (1996) was cited in the position paper: “In general terms, EM addresses the systematic analysis and modelling of processes, organization structures, product structures, IT-systems or any other perspective relevant for the modelling purpose.” This definition of EM implies that EM may or may not describe an entire enterprise, may cover some of the same territory as SA&D (systems analysis and design) and BPM (business process management), and may or may not be directly related to IT.

**Separation between WST/WSM and enterprise modelling.** Although Bock et al. (2014) included WST along with Archimate, DEMO, and MEMO in its comparison of four enterprise modelling approaches, to date WST/WSM generally has not been viewed as part of the EM discourse for several

reasons: First, WST/WSM focuses explicitly on analysing and designing work systems, a conceptual lens that is not widely recognized in the EM community. It could be used for EM, however, because any given enterprise can be viewed as consisting of multiple work systems that can be analyzed and designed using WST/WSM. Second, models created to date by using WST/WSM have been more informal than models produced by using more formalized EM approaches such as BPMN, Archimate, MEMO, and DEMO. According to Bork and Fill (2014), the foundations of formal modelling methods include the modelling language, modelling procedure, and mechanisms and algorithms. A modelling language is described in terms of its semantics, syntax, and notation (Karagiannis and Kühn, 2002). WST/WSM addresses semantics by defining many terms carefully, but currently does not have a formal language with defined syntax and notation. This paper describes initial steps in that direction.

**A thought experiment.** Thought experiments have been used or suggested in IS research for at least two decades. For example, Introna and Whitley (1997) cite Kuhn's recognition of the importance of thought experiments in the physical sciences and identify thought experiments by Searle, Introna, and Varela that were relevant to IS. This paper describes a thought experiment that uses a hypothetical situation as a way to imagine the form and content of an initial version of an EM language. We view the thought experiment as a step toward a broad and ambitious research question:

*RQ: Develop a modelling language (with defined semantics, syntax, and notation) that is based on WST/WSM and is useful for work system modelling and for enterprise modelling.*

Using a thought experiment is somewhat outside of established IS methodologies. It is more in line with an "alternative genres" approach highlighted in a special issue the *European Journal of Information Systems* (Avital et al., 2017). It is somewhat in the spirit of Grover and Lyytinen's (2015) call for new types of knowledge production in IS and with Burton-Jones et al. (2015) and Ortiz de Guinea and Webster (2017) calling for combining existing perspectives or creating hybrid approaches.

**Goal and organization.** This research-in-progress paper explains initial progress in pursuing the RQ. The next section mentions WST/WSM briefly, but focuses on a work system metamodel, an extension of WST that reinterprets the concepts in the work system framework in a more detailed form. A reduced version of the metamodel called the minimalist metamodel (MMM) is proposed as the basis of a thought experiment. Imagine an enterprise modelling language (the MMM language) that is based on MMM. Imagine that the MMM language is used to create an enterprise model that would help software vendor X, which is involved in an internal reengineering project. Does it seem plausible that the MMM language could be useful for analysing reengineering possibilities in that project? Based on a partial specification of the enterprise model, it seems possible that the MMM language and/or variants of it could be useful.

## **2 Work System Theory, Work System Method, and Work System Metamodel**

The core of WST/WSM is the assumption that the topic at hand is a work system and that work systems can be understood, analysed, and designed using WST, whose three components were designed to be straightforward enough to avoid seeming overwhelming to business professionals and researchers who need to think about a work system in an organization but do not need level of detail approaching detailed requirements for software development (consistent with Sandkuhl et al. 2016).

**Work system theory.** The three components of WST are the definition of work system, the work system framework (WSF), and the work system life cycle model (WSLC). A work system is defined as "a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce specific product/services for internal or external customers." (An IS is a type of work system most or all of whose activities focus on processing information, i.e., capturing, transmitting, storing, retrieving, manipulating, and/or displaying information.) The WSF identifies and organizes nine elements of even a rudimentary understanding a work system's form, function, and environment during a period when it is relatively

stable even though incremental changes such as minor personnel substitutions or technology upgrades may occur within what is still considered the same work system. Its elements include customers (of the work system), product/services (produced by the work system), processes and activities, participants, information, technologies, environment, infrastructure, and strategies. The WSLC model represents the iterative process by which work systems evolve over time through a combination of planned change (formal projects) and unplanned (emergent) change that occurs through adaptations and workarounds. The WSF and WSLC have been published many times (e.g., Alter, 2006, 2013) and will not be presented in detail here.

**Work system method.** Applications of WST in teaching have used various versions of the work system method (WSM) that all embody the same “way of working” (Rolland et al. 1995; Bock et al. 2014) individually or in collaboration with business stakeholders and/or IT professionals: Identify the main problems or opportunities; identify the smallest work system that has those problems or opportunities (plus relevant constraints, key incidents, and so on); use the work system framework to summarize the “as is” work system; analyze the situation to whatever depth is needed; recommend a proposed “to be” work system; explain why the proposed IT-enabled work system is likely to exhibit better performance than the existing work system. (Alter, 2006, 2013). A key assumption is that improvements in the work system’s technology without changing other aspects of the work system rarely have a major impact unless the main problem was deficient technology.

**Need for a metamodel.** WST/WSM provides a relatively informal approach for understanding and analyzing systems that includes useful guidance about what to consider and how to proceed in an analysis. It does not use formal notation, syntax, or semantics that characterize modelling languages (see Karagiannis and Kühn, 2002; Bork and Fill, 2014). The need for additional guidance in the form of a metamodel became apparent when users of the work system framework found it very helpful for discussing a work system’s scope, but not as helpful for explaining in detail exactly what information and other resources are used by each step in a business process. Most of those users were MBA and Executive MBA students who were analyzing work systems in their own organizations in order to produce management briefings that proposed improvements in those systems. (e.g., Truex et al., 2010). Examination of their analyses over around a decade led to concluding that a formal metamodel might help. To date, six successive versions of a work system metamodel have been produced, with the most recent appearing in Alter, 2016; Alter and Bolloju (2016). All of those metamodels express the ideas in the work system framework in more detail and in a graphical form that brings work system ideas closer to the type of formality that fits with the form and spirit of much of the EM research and practice to date. Major parts of the metamodel are summarized in Table 1; other parts are omitted.

**Enterprises and value constellations** (business ecosystems) consist of **work systems**.

**Work systems** always contain at least one **activity**. They may contain one or more **business processes** if a set of **activities** is sufficiently interrelated and sequential to qualify as a process.

**Activities** use **resources** to produce one or more **product/service from activity** that may be used as a **resource** for subsequent **activities** and/or may contribute to a **product/service offering** for a customer. A **product/service offering** may combine multiple **product/services from activities**. Thus, only some of the **product/services from activities** are included in **product/service offerings** that are received or used by **customer work systems**, and hence by **customers**.

**Customer work systems** create **value for customers** using their own resources plus **product/service offerings** produced by the provider **work system**.

**Resources** used by an **activity** may include human resources (**participants**), **informational resources**, **technological resources**, and **other resources**. The metamodel includes specific types within each resource type to minimize the likelihood of omissions in an analysis.

**Activities** are performed by **actor roles** that can be performed by three types of

entities, **noncustomer participants**, **customer participants**, and **encapsulated services**. In medicine, a doctor is a **noncustomer participant**, the patient is a **customer participant**, and software that automatically identifies drug interactions is an **encapsulated service**.

**Encapsulated services** may be **automated services** such as (totally automated) search engines or sociotechnical services that happen to be encapsulated, such as performing pathology evaluations.

The outcome of **activities** that use human resources (**participants**) depends on **participant** attributes, such as **knowledge/ expertise, skills/ capabilities, performance metrics**, and **motives**.

**Technological resources** used in an **activity** may include **tools** used directly by **participants** (e.g., a truck) or **automated services** that operate autonomously after being launched (e.g., a search engine).

Both the **work system** and **customer work system** may interact with **other work systems** with positive and/or negative impacts on the operation of either work system.

Table 1. Summary of major parts of the sixth version of a work system metamodel

A graphical representation including all of the metamodel's 50+ entity types is omitted because it was published elsewhere (noted above) and would absorb one of this paper's 7 pages. Its complicated nature results from trying to serve as a reminder of important issues that might be overlooked, such as the fact that activities can be performed by customer participants (e.g., doctors or teachers) and noncustomer participants (e.g., patients or students) and that technologies can take the form of tools or automated services. While there is no reason to assume that an enterprise model based on the metamodel will include all of those entity types (just as there a BPMN or UML model typically will not use all of their features), the complicated appearance of the metamodel may seem overwhelming.

The current research explores the feasibility of starting again with a much simpler metamodel that overcomes the non-operational nature of the WSF but is less complicated than the existing work system metamodels. In essence, the WSF represents an implicit metamodel saying that each of nine entity types is linked directly to the single entity type **work system** by a relationship that might be called "part of a rudimentary understanding." An operational metamodel needs to contain relationships that describe how a work system operates. (The arrows within the graphical representation of the WSF are reminders of the need for mutual alignment and do not describe operational interactions.)

A "**minimalist metamodel**" (MMM). A reduced version of the work system metamodel is proposed here as a step toward applying WST/WSM in an enterprise model that might be the basis of enterprise simulations or other automated analysis. The work system metamodel was created as a step toward the level of specificity required for programming, but was still in the spirit of WST, i.e., focusing on things that business and IT professionals might want to think about in an organized way. It did not try to provide a direct, computerized link to automated simulation or analysis of work systems or enterprise models based on the metamodel. A far simpler metamodel might be a step in that direction. Instead of trying to remind business and IT professionals about the many situationally important topics that might be organized using a metamodel (as in Table 1), a much reduced metamodel might be more effective as the basis of an initial, test version of a WST-based enterprise modelling approach that would avoid excessive effort in dealing with too many concepts. A drastically reduced metamodel will be called the "minimalist metamodel (the MMM) to avoid confusion with the complete version. The MMM is proposed as the smallest metamodel that might be used meaningfully.

Assume that the MMM is to be used in enterprise modelling focusing on basic issues related to enterprise reengineering, such as division of labor and ways in which usage of essential resources by some work systems tend to interfere with use of the same resources by other work systems. The MMM for an enterprise model that could be simulated would include the following: It would be based on a meta-metamodel saying that any entity type would have zero or more attributes that may be **characteristics, goals, metrics, and performance measurements** (measured at a time  $t$  or across a specified time interval). Table 2 presents a possible version of the MMM in textual form.

An **enterprise** consists of one or more **work systems**.

A **work system** always contains at least one **activity** and always produces, changes, or otherwise affects one or more resources for the benefit of one or more **customers**.

The four types of resources are **human resources**, **technical resources**, **informational resources**, and **other resources** (such as time, money, and product/services that go to another activity)

**Activities** use one or more resources (**human resources**, **technical resources**, **informational resources**, and/or **other resources**) to produce, change, or otherwise affect resources that may include the resources they use (e.g., converting an isolated engine into a part of a car by assembly).

An **activity** occurs during a time interval. For example, an **activity** could last four months, could occur three times during a day, or might not occur at all during the time of interest. The characteristics start time and end time that are associated with the activity in general could be recorded whenever an activity occurs.

Table 2. A possible version of the MMM metamodel.

**Semantics, syntax, and notation of the MMM language.** Recall that modelling languages can be described in terms of their semantics, syntax, and notation. Whereas the semantics of the sixth version of the metamodel involves over 50 entity types, the MMM contains only 8 entity types: enterprise, work system, activity, human resource, informational resource, technical resource, other resource, and customer (remembering that the meta-metamodel says that entity types may have characteristics, goals, or other attributes). The MMM eliminates the many subcategories of human, informational, technical, and other resources included in the complete metamodel in Table 1. Those subcategories can help a user think about a specific work system but tend to make an enterprise model unnecessarily complex. The MMM contains 5 types of relationships: *contains* (e.g., work system contains activities or contains work systems as subsystems), *uses* (e.g., activity uses resource), *produces* (e.g., activity produces or changes state of a resource), *benefits* (e.g., a resource produced by an activity benefits one or more customers), and *interacts with* (e.g., a work system interacts with a work system).

The syntax and notation of the MMM language need to represent the 8 types of entities and 5 types of relationships. The syntax was summarized above in terms of the possible relationships between entity types. The notation should be adapted from familiar notations from conceptual modelling. For example, activity could be represented using a rectangle and informational resource might be represented using a symbol related to information. The relationships could be represented using typical line and arrow conventions from conceptual modelling.

### 3 Hypothetical Example for a Thought Experiment

Consider software vendor X that needs to change its way of working with its clients. The operation of Development, Sales, and Service departments at Vendor X will be viewed as separate work systems that are the focus of a reengineering effort. Currently, Development produces software that is delivered to customers. Sales performs sales activities that end with installation of the software. Service helps with customer problems and provides some consulting.

Customers are finding this segmentation of duties increasingly unsatisfactory. Customers want to deal with Solution teams that are responsible for helping them make the software successful in customer sites. This will require a major reorganization, with the software firm operating more along the lines of DevOps (Roche, 2013), i.e., developers and technical support staff take more direct responsibility for customer success with the software, including both customization and cloud-based operation.

**Representation based on the MMM language.** In analysing the existing system and proposed changes, Vendor X uses the MMM language to produce an enterprise model. The enterprise model tries to represent the new way of working within Vendor X. The enterprise model contains separate submodels for the three departments, but many key participants in each submodel are also participants in other submodels. For example, high level developers often will participate in sales calls to assure

that needed customization is possible and later will participate in customer service activities if customers encounter software-related difficulties that interrupt their production processes.

An excerpt from the imagined conceptual enterprise model for Vendor X is represented in a tabular form in Tables 3 and 4. Table 3 is a highly summarized representation of the Development work system in the newly proposed way of working. The columns in Table 3 are based on the semantics of the MMM language, e.g., activities, resources, characteristics, and goals. Much other information would be included in a more complete table or a set of related tables. For example, each activity uses or produces informational resources that are not mentioned in Table 3, such as triggers, preconditions, postconditions, start time, and end time. Notice how business rules can be listed under resources used or could be treated as a separate type of informational resource. Similar activity/resource tables hypothetically exist for the other two work systems that are being considered.

Activity	Resource used	Resource produced	Characteristic	Goal	Performance criterion	Business rules
Design software	Programmer Designer Product extension requirements	Design of software, Explanation of design decisions	Complex, Requires expert skills	Architectural compliance, Complete explanation	Satisfaction of management Feasibility assessment by programmers	(Could have multiple rules for each activity).
Program software	Programmer, Requirements from design effort	Completed, debugged programs, Documentation	Semi-complex Requires average knowledge	Standard rate of output, Standard bug tolerance	Comparison with standard output rate, Bug rate	(Could have multiple rules for each activity).
Customize software	Programmer, Customer representative, Customer need	Completed, customized program	Complex, Requires expert	Acceptance of specification, Acceptance of result	Customer satisfaction, Low follow-up requirement	(Could have multiple rules for each activity).
Provide software support	Programmer, Customer caller Explanation of problem	Consultation, Customer understanding, Documentation of resolution	Tier of support (1, 2, or 3), Style of support	Customer satisfaction with result, Customer knowledge	Rate of resolution on first call, Total time by type of call	(Could have multiple rules for each activity).
Bill for software support	Time sheet of programmer, Contract	Bill for customer	Time charged, Rate, Total bill	Accuracy of bill, On time delivery	Accuracy of bill, On time delivery	(Could have multiple rules for each activity).

Table 3. Illustration of how entity types in the minimalist metamodel (MMM) can be used to model activities and related resources in the Vendor X example.

The first column of Table 4 shows the four types of resources in the MMM. The subtypes in the next column come from the complete metamodel. In an enterprise model based on the MMM these subtypes can be handled as characteristics that apply to specific instances of the four types of resource. For example, a human resource (a work system participant) may have the characteristic *non-customer participant*, which would apply to a developer, or the characteristic *customer participant*, which would apply to a customer calling for help. Other representative examples are included in Table 4.

**Instantiation of the enterprise model.** The concepts in Tables 3 and 4 are part of a conceptual model based on the MMM that is being used to understand the implications of a proposed new way of working at Vendor X. An instantiation of that model would require filling in the detailed information for specific (but hypothetical) activities, developers, customers, and so on. Some of the resources (especially “human resources”) would be dedicated to an activity or department. Other resources would be shared across activities or departments if the new way of working is adopted. One of the reasons for performing the analysis is to decide whether the proposed change would be feasible and would foster an equitable work environment. Producing that information to match the proposed situation at Vendor X would require a substantial effort, but that type of effort is needed in order to

perform a reasonably realistic simulation based on the enterprise model. The same type of information would need to be available to perform queries and calculations, such as the percentage of developer time that will be devoted to customer services with the new way of working.

Type of resource	Resource subtype	Characteristic	Goal	Ownership or affiliation	Sharing	Corporate policies
Human resource	Non-customer participant, Customer participant,	Age Knowledge level, Skill level,	Output goal by job spec, Goal for skill development, Knowledge sharing goal	Department, Professional certifications	Work systems in which this person participates	Employee manual, Union contract,
Technological resource	Tool, Automated service	Hardware type Software type On premises or in cloud	Uptime, Response time, Total cost of ownership	Department that manages the asset (including infrastructure)	Work systems that use this technological resource	Transfer pricing policy, Policy on network usage
Informational resource	Transaction record, Plan, Forecast,	Precision, Complexity, Source, Security	Accuracy, Timeliness, Reliability	Department that manages information (including infrastructure)	Work systems that use this informational resource	Information security policy,
Other resource	Time, Culture, Product/service	Produced internally or externally	(Various goals for different types of “other resources”)	Department or other owner, wherever relevant	External users of the resource, wherever relevant	General policy on usage of corporate assets

Table 4. Examples showing the information related to different types of resources

## 4 Interim Conclusions

This paper showed that a minimalist metamodel (MMM) appropriate for enterprise modelling can be produced by pruning many of the concepts in a published version of a work system metamodel. Much of the reduction occurs by treating entity types such as *customer participant*, *plan*, *tool*, or *culture* as subtypes of the human, informational, technological, or other resources. Reducing the number of entity types and relationships to just 8 entity types (not including characteristics, goals, and so on from the meta-metamodel) and 5 relationship types makes the MMM easier to use directly as the basis of an MMM enterprise modelling language. The semantics and syntax of the language are easy to express in natural language. Notation was not included due to this paper’s 7-page length limit, but can be imagined easily based on widespread use of BPMN, Archimate, and other diagram-oriented tools.

**Next steps.** Moving forward from this point calls for creating other hypothetical situations and exploring the extent to which MMM and the MMM language address those situations in a meaningful way, i.e., in a way that leads to insights either through conceptual models or through instantiations.

Overall, the MMM and the MMM language should be viewed as one among many possibilities for adapting the work system metamodel for enterprise modeling. For example, the MMM language includes technological resources, but basically treats technology as a tool that is used in an activity. More fully covering the application (software) and technology (infrastructural services) layers in Archimate calls for a different approach that can be accomplished by adding *encapsulated service* (which exists in the full metamodel) as an entity type. Many other variants could be valuable, such as including *environment* or including *customer work systems* as is done in the full metamodel because value is co-created when provider activities coincide with the customer’s value creating activities (e.g., see Grönroos, 2011 and Vargo and Lusch, 2016). An additional project might focus on criteria for justifying successive incremental inclusions of additional entity types.



## References

- Alter, S. (1995). “How should business professionals analyze information systems for themselves?” pp. 284-299 in E. Falkenberg, et al, *Information System Concepts: Toward a Consolidation of Views*, Proceedings of the IFIP WG 8.1 Working Conference on Basic Information System Concepts, Marburg, Germany, 1995.
- Alter, S. (2006) *The Work System Method: Connecting People, Processes, and IT for Business Results*, Larkspur, CA: Work System Press.
- Alter, S. (2013). “Work System Theory: Overview of Core Concepts, Extensions, and Challenges for the Future,” *Journal of the Association for Information Systems*, 14 (2), 72-121
- Alter, S. (2016). Encapsulation as a key concern in analysis and design for service systems. In *Proceedings of the Americas Conference on Information Systems*.
- Alter, S., and Bolloju, N. (2016). “A Work System Front End for Object-Oriented Analysis and Design.” *International Journal of Information Technologies and Systems Approach (IJITSA)*, 9(1), 1-18
- Avital, M., Mathiassen, L., and Schultze, U. (2017) “Alternative genres in information systems research,” *European Journal of Information Systems*, 26(3), 240-247.
- Bock, A., Kaczmarek, M., Overbeek, S. and Heß, M., (2014) “ A Comparative Analysis of Selected Enterprise Modelling Approaches,” 7th IFIP Working Conference on The Practice of Enterprise Modelling (PoEM), Springer, Lecture Notes in Business Information Processing, LNBIP-197, pp.148-163, 2014, The Practice of Enterprise Modelling.
- Bork, D. and Fill, H.G., (2014). “Formal aspects of enterprise modeling methods: a comparison framework.” In *System Sciences (HICSS), 2014 47th Hawaii International Conference on* (pp. 3400-3409). IEEE.
- Bolloju, N., Alter, S., Gupta, A., Gupta, S. and Jain, S. (2017). “Improving Scrum User Stories and Product Backlog Using Work System Snapshots.” In *Proceedings of the Americas Conference on Information Systems*.
- Burton-Jones, A., McLean, E.R. and Monod, E. (2015). “Theoretical perspectives in IS research: from variance and process to conceptual latitude and conceptual fit.” *European journal of information systems*, (24:6), 664-679
- Grönroos, C. (2011). “Value creation in service logic: A critical analysis,” *Marketing Theory*, 11 (3), 279-301.
- Grover, V. and Lyytinen, K. (2015). New State of Play in Information Systems Research: The Push to the Edges, *MIS Quarterly*, 39(2), 271-296.
- Introna, L.D. and Whitley, E.A., (1997). “Imagine: thought experiments in information systems research,” In *Information Systems and Qualitative Research* (pp. 481-496). Springer US.
- Karagiannis, D. and Kühn, H., (2002) “Metamodelling platforms.” In *EC-Web* (Vol. 2455, p. 182).
- Ortiz de Guinea, A.O., and Webster, J. (2017) “Combining variance and process in information systems research: Hybrid approaches. *Information and Organization*,” 27(3), 144-162
- Roche, J. (2013). Adopting DevOps practices in quality assurance. *Communications of the ACM*, 56(11), 38-43.
- Rolland, C., Souveyet, C., and Moreno, M. (1995). “An approach for defining ways-of-working.” *Information Systems*, 20(4), 337-359.
- Sandkuhl, K., Fill, H.G., Hoppenbrouwers, S., Krogstie, J., Leue, A., Matthes, F., Opdahl, A.L., Schwabe, G., Uludag, Ö. and Winter, R. (2016). Enterprise Modelling for the Masses – From Elitist Discipline to Common Practice. In *IFIP Working Conference on The Practice of Enterprise Modelling* (pp. 225-240). Springer International Publishing.
- Sandkuhl, K., Fill, H.G., Hoppenbrouwers, S., Krogstie, J., Leue, A., Matthes, F., Opdahl, A.L., Schwabe, G., Uludag, Ö. and Winter, R. (2018). “From Expert Discipline to Common Practice: A Vision and Research Agenda for Extending the Reach of Enterprise Modeling.” *Business & Information Systems Engineering*, 60(1), 69-80.

- Truex, D., Alter, S. and Long, C. (2010). “Systems analysis for everyone else: Empowering business professionals through a systems analysis method that fits their needs.” *European Conference on Information Systems*.
- Vargo, S.L. and R.F. Lusch (2016). “Institutions and axioms: an extension and update of service-dominant logic,” *Journal of the Academy of Marketing Science*, 44(1), 5-23.
- Vernadat, F. B. (1996). *Enterprise Modelling and Integration*. Chapman & Hall.