Work System Modeling Method with Different Levels of Specificity and Rigor for Different Stakeholder Purposes

Steven Alter[1] and Dominik Bork[2]

[1] School of Management, University of San Francisco, San Francisco, USA
alter@usfca.edu
[2] University of Vienna, Faculty of Computer Science, Vienna, Austria
dominik.bork@univie.ac.at

Abstract. This paper proposes a modeling method (the work system modeling method - WSMM) that addresses key issues related to enterprise and process modeling. Those issues lead to modeling method requirements that call for relaxing common assumptions about the nature of modeling methods and related modeling languages and metamodels. A summary of work system theory (WST) and the work system method (WSM) provides background for understanding WSMM. A design space for modeling positions most applications of WSM in relation to seven purposes of modeling that call for successively more formal approaches. WSMM is presented in relation to the seven purposes, thereby extending WSM in new directions. A final section summarizes how WSMM addresses the issues and requirements from the introduction, explains how coherence is maintained within WSMM, and identifies areas for future research.

Keywords: modeling method, modeling language, work system method, work system theory

1 Overcoming Problems in Enterprise and Process Modeling

Prominent researchers argue from various backgrounds that modeling methods for enterprise and process modeling have not achieved their full potential and need to be extended or augmented to make them more usable by broader user groups. This paper's approach to modeling methods addresses important issues that they mention:

- A Dagstuhl seminar [1] emphasized how differing stakeholder needs call for different approaches to enterprise modeling (EM). That seminar led to a *BISE* research note by EM community leaders [2] that encourages moving EM from an expert discipline towards "grass roots modeling" and "modeling for the masses." Their future research agenda includes "softened requirements to completeness, coherence and rigor".
- Six of seven process modeling problems discussed in [3] are relevant here: 1) aiming for one model that suits all purposes, 2) straightjacketing smaller interactive processes into one monolithic model, 3) using static hierarchical decomposition as the only abstraction mechanism, 4) modeling humans as if they are machines doing

14th International Conference on Wirtschaftsinformatik,
February 24-27, 2019, Siegen, Germany

a single task, 5) being vague about vagueness, 6) abstracting [away] from the things that really matter [to stakeholders].

- [4] calls for "overcoming tendencies to view diagrammatic modeling methods and languages" as "stable, even standardized, artefacts that establish some commonly agreed way of describing a 'system under study,' [which] implies that all stakeholders work on the same level of abstraction and specificity."

Related research on modeling method usage (e.g., [5, 6]) and model comprehension (e.g., [7–9]) illuminates major issues. Many modelers do not apply modeling methods as intended by their designers, frequently using only a subset of the syntactic concepts provided [10]. Modeling methods often do not fit modelers' aptitudes, knowledge, and purposes [11, 12]. [13] notes that the "lack of intuitiveness of diagrammatic representations and the complementary role of text-based representations has been underlined in recent research." Cognitive load [14] for stakeholders becomes increasingly important as unfamiliar symbols and icons proliferate. [13] also mentions lack of flexibility in process models, dilemmas of control, and excessive prescriptiveness. Uncertainty and variability related to accidents, mistakes, and intentional workarounds bring further challenges for modeling methods.

Requirements for a More Flexible Modeling Method. This paper pursues four requirements by presenting a modeling method that relaxes many common assumptions that are obstacles to dealing with those issues about modeling and modeling methods.

1. The modeling method should respect stakeholder diversity related to knowledge, beliefs, and roles, thereby making it usable both by business professionals working individually and in collaboration and by IT professionals pursuing model-driven development or code generation. (cf. [5]).
2. A modeling method can include different modeling techniques for different stakeholder purposes related to the same situation (contrary to the view in [15] that a modeling method can have only one modeling technique that combines a single modeling language and a modeling procedure).
3. With different modeling techniques for different purposes, a modeling method can use different modeling languages based on different metamodels. In relation to domain-specific conceptual modeling (cf., [16]), this approach assumes that intersubjective understanding between stakeholders might not require a single metamodel for processes, services, enterprises, goals, and so on.
4. The representation of a model might or might not use diagrams with rigorously defined notation and syntax (e. g., BPMN, ArchiMate) or might use such diagrams for some purposes but not for others.

Acceptance of multiple techniques, modeling languages, and metamodels within a modeling method leads to challenges about maintaining coherence across different models produced by different stakeholders for different purposes. Our approach to coherence within a modeling method is to require use of a single overarching metaphor that applies to all modeling techniques of the modeling method. According to [17, p. 138], a modeling method metaphor defines a specific perspective taken by the modeler while observing the reality and mapping the relevant aspects to the modeling language at hand, thereby creating a model representation of the reality. Ideally that metaphor should help in bridging gaps between modelers and practitioners who often visualize situations from different viewpoints. The invariance of the single overarching metaphor ensures that all modeling techniques contribute to an overall goal, even if they employ different levels of detail and expressiveness. Differences between the models will be revealed in stakeholders' personal understandings and collaborative discussions.

A Work System Modeling Method. This paper extends several decades of effort related to the work system method (WSM), a flexible systems analysis and design approach based on an informal type of modeling and problem-solving designed to help business professionals visualize work systems and collaborate more effectively with IT professionals [18–21]. Many hundreds of MBA and Executive MBA students have used various versions of WSM templates that guided their production of management briefings about improving real world work systems. Those templates contain many modeling techniques that have never been expressed as a formal modeling language (cf. [22]). For example, none is based on an explicit metamodel or operationalized along a procedure.

Research Goal and Organization. This paper explains how a work system modeling method (WSMM) based on WSM, WST, and a central work system metaphor satisfies the four requirements above. WST and WSM provide a plausible starting point for developing WSMM because their spirit is aligned with the "modeling for the masses" vision in [2] and because an enterprise is a set of interacting work systems.

Building on our previous work [23], this paper provides contributions in several areas by showing that WSMM could help modelers and users apply a range of modeling techniques to situations that seem difficult to address without relaxing assumptions such as use of a single modeling technique, a formal modeling language, and diagrammatic notation. WSMM is a step toward a modeling tool that can be implemented using existing metamodeling platforms such as ADOxx. Reflection on WSMM's form and assumptions might help researchers and practitioners wishing to revise or design modeling methods including those for domain-specific conceptual modeling. It might contribute to reflection on how modeling standards such as BPMN, EPC, and ArchiMate can be adapted to address needs of broader audiences. It also might show a path for formalized extensions of less structured contexts such as those addressed by Checkland's soft systems methodology [24].

The next section summarizes WST and WSM to introduce the central work system metaphor. A two-dimensional design space for modeling methods illustrates the context of WSM. One dimension traverses seven purposes of modeling that require a range of modeling techniques from quite informal to highly formal (cf. [22]). A second dimension represents different degrees of specificity in content, appearance, and usage. The subsequent section illustrates the scope of WSMM using an example that emphasizes modeling techniques and related metamodels. A final section summarizes how WSMM addresses the requirements mentioned above, explains how WSMM maintains coherence across models built for different purposes, and identifies challenges for future research.

2 Thinking of Systems in Organizations as "Work Systems"

This section provides background related to the work system method (WSM), an informal modeling approach that to date has not been guided by a metamodel and that does not produce specifications in the sense of enterprise modeling.

Work system basics. A work system is a system in which human participants and/or machines perform processes and activities using information, technology, and other resources to produce product/services for internal and/or external customers. The and/or in the definition implies that work systems can be sociotechnical (with human participants) or totally automated. A work system operates within an environment that matters (e.g., national and organizational culture, policies, history, competitive situation, demographics, technological change, other stakeholders, and so on). Work systems rely on human, informational, and technical infrastructure that is shared with other work systems. Work systems should support enterprise and departmental strategies. The definition of work system specifies that work system is a very general case that includes many special cases such as information systems, supply chains, service systems, projects, and totally automated work systems. For example, an IS is a work system most of whose activities are devoted to processing information. Supply chains are work systems that extend across multiple organizations to provide resources for other organizations. Projects are work systems that produce specific product/services and then go out of existence. An enterprise or organization is a series of interacting work systems.

WST, the theoretical basis of WSM, consists of three parts: 1) the definition of work system, 2) the work system framework, and 3) the work system life cycle model, which is not discussed here. This paper makes direct use of the definition and of the work system framework (Figure 1), which outlines elements of even a rudimentary understanding of a work system's form, function, and environment as the work system exists during a time interval when its structure is basically static. Emphasizing business rather than IT concerns, this framework covers situations that might not have a well-defined business process and might not be IT-intensive. Processes and activities, participants, information, and technologies are viewed as completely within the work system. Customers and product/services may be partially inside and partially outside because customers often participate in work systems. A common limit to modeling precision is that human participants in work systems may make errors and may pursue adaptations and workarounds instead of following prescribed procedures. Furthermore, processes fall along a dimension from unstructured to structured [25], starting with largely unstructured creative processes (such as many design and management processes) that have no pre-specified sequence, may involve extensive iteration, and therefore are not amenable to detailed, high precision modeling.

Work system method. WSM is a semi-formal systems analysis and design approach that was developed over several decades to help business professionals visualize work systems in their own organizations and collaborate more effectively with IT professionals. To date, almost all use of WSM has applied work system analysis templates that outline how to proceed from aspects of a work system's structure and performance toward producing a preliminary recommendation about how to improve the work system. The templates include some questions that require textual answers, others that require filling out formatted tables, and others that invite users to include swimlane diagrams, Pareto charts, or other diagrams if they have appropriate software.

While details differ, every version of WSM is organized around the following: 1) identify the smallest work system that has the problem or opportunity; 2) summarize the "as-is" work system using a work system snapshot (example in Table 2), a stylized one page summary; 3) evaluate work system operation using measures of performance, key incidents, social relations, and other factors; 4) drill down further as necessary; 5) propose changes by producing a work system snapshot of a proposed "to be" work system that will probably perform better; 6) describe likely performance improvements.

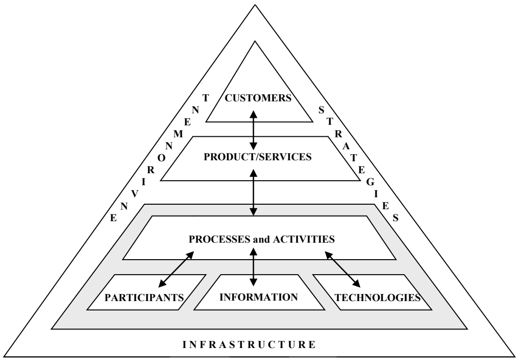

Figure 1. Work system framework [19, 20]

3 Framework for Visualizing Modeling Methods

We agree with the view in [2] that "not all knowledge should be represented as a formal model" and that it is important to find "the right balance of representational forms," including formal and informal models. A clear discussion of this entire topic requires a foundation such as the framework in [15] by which a modeling method is a composition of a *modeling language*, *modeling procedure*, and *mechanisms & algorithms*. A modeling language, the backbone of a modeling method, is composed of three components: syntax (the concepts provided by a modeling language, including their valid combinations), semantics (the meaning of the concepts), and notation (the graphical representation of the concepts). The combination of a modeling language with a modeling procedure is referred to as a *modeling technique*.

An alternative perspective on modeling methods. Facilitating modeling by diverse stakeholders calls for relaxing the requirement in [15] that all stakeholders and purposes must be accommodated using a single modeling technique, i.e. one modeling language and one modeling procedure. Relaxing that requirement avoids cognitive overload from increasing metamodel complexity due to adding concepts to separate modeling techniques when diverse stakeholder needs and purposes call for them. Thereby, the cognitive load of each single modeling technique is minimized by maintaining the overall expressiveness of the modeling method.

An alternative view of modeling methods starts with a modeling method design space. The ideas explained next are equally applicable to design spaces based on the work system metaphor or other central metaphors such as systems in general, sociotechnical systems, actor networks, activity theory, and viable systems.

3.1 Design Space for Modeling Methods and Modeling Techniques

Figure 2 represents a design space for modeling methods and modeling techniques related to a core metaphor. A key goal of the design space is accommodating a range of stakeholder purposes, shown as P1 through P7. Technique specificity is the extent to which a technique defines exactly what to include, what to ignore, and how to proceed. Techniques with low specificity tend to be flexible but at the cost of providing relatively little conceptual or procedural guidance. The reverse applies as well.

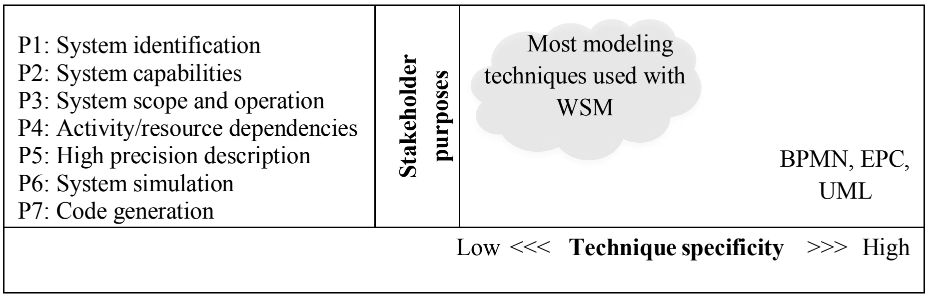

Figure 2. Design Space for Modeling Methods and Modeling Techniques

The shaded area represents the positioning of most of the modeling techniques that WSM users have applied. Most of those techniques focus on topics such as work system *scope and operation*, and *activity/resource dependencies*. Those techniques are relatively low in specificity compared to techniques that might be used for high precision description, system simulation, or code generation. Instead of accepting the assumption that WSMM would include only one technique, this paper assumes that WSMM could include techniques anywhere in the design space provided that those techniques genuinely fit with the overarching metaphor and stakeholder purposes.

4 Work System Modeling Method

WSMM expands WSM greatly by recognizing a wide range of stakeholder purposes and different degrees of specificity for different stakeholders that all use the central work system metaphor. WSMM provides a broader scope of modeling to help various users understand the situation and decide how to improve it. WSMM uses simpler metamodels for informal and intuitive visualization of work systems and more precise and expressive metamodels for helping decision makers identify and select among possible changes and for helping software developers produce or improve software.

WSMM is a highly adaptable method organized around a set of modeling techniques based on a work system metaphor. That range of techniques should be available where required by a range of stakeholders who may encounter situations that resist being modeled within restrictions that are often viewed as essential to conceptual modeling. Our WSMM test case is a hiring system. Similar test cases could be constructed for work systems in production, sales, accounting, software development, and so on.
Managers of an engineering firm are concerned that their current work system for finding, interviewing, and hiring job applicants takes too long, wastes too much effort in interviews, and seems to hire too many engineers whose contributions to engineering projects are disappointing. Some managers believe that a better online human resources (HR) portal would help. Others believe that the problems lie elsewhere.

4.1 Modeling Techniques within WSMM

WSMM should include modeling techniques that support different stakeholder purposes identified in Figure 2. The following discussion uses the hiring example to illustrate the scope of an initial version of WSMM in relation to purposes P1 through P7, but not to show all imaginable modeling techniques that might be included eventually, and certainly not to go into a lengthy explanation of specific techniques.

P1: Identification of the work system. Simply naming the work system of interest with a verb phrase such as *"finding, interviewing, and hiring applicants"* often avoids confusion with people thinking that the technology (e.g., the online HR portal) is the primary object of the improvement effort. Clarity in that regard makes sure that the project is viewed as much more than a software project that produces software. P1 makes no attempt to describe the work system's behavior and structure and does not call for a specific procedure. The metamodel consists of one concept: *Work System.*

Table 1. Two illustrations of lists of capabilities (P2) for the hiring example

Table 1a: Simple list of capabilities	**Table 1b: List of capabilities with performance or service level expectations**
• defining parameters of the position • publicizing the position • prioritizing applications	• defining parameters for a position, responding within 3 days of request • publicizing the position, advertising in at least five bulletin boards • prioritizing applications, responding within three days of due date

P2: Capabilities of the work system. The hiring work system has capabilities that can be included as a service catalog explanation for the P1 work system description. Those capabilities might be described using a list of verb phrases such as those in Table 1a. A slightly more complete description of capabilities might include performance or service level expectations for each capability in Table 1b. A minimalist metamodel for P2 includes the concepts *Work System* and *Capability* and a slightly more detailed metamodel would add *Performance expectation* or *Service level*. The P2 capabilities lists in Table 1 are simple in form. Again, a specific procedure is not employed.

Researchers focusing on capability driven-development have used a more rigorous notion of capability, e.g., the 15 concepts in the capability-related metamodel in [26].

P3. Scope and general operation of the work system. With P3, the stakeholder wants to clarify the scope of the work system (based on P1) and to attain an overview of its general operation without going into great detail. Table 2 illustrates P3 in the form of a "work system snapshot", a modeling technique for describing a work system's scope and general operation. This is a formatted one-page summary of a work system in terms of the six central elements of the work system framework. Those six elements provide an easily used description that helps in defining the boundaries and contents of the work system that has the problem or opportunity at hand.

Table 2. Illustration related to P3: A work system snapshot of a hiring work system

<table>
<tr><th colspan="2">Customers</th><th colspan="2">Product/Services</th></tr>
<tr><td colspan="2">• Hiring manager
• Larger organization (which will have the applicant as a colleague)
• HR manager (who will analyze the nature of applications)</td><td colspan="2">• Applications (which may be used for subsequent analysis)
• Job offers
• Rejection letters
• Hiring of the applicant</td></tr>
<tr><th colspan="4">Major Processes and Activities</th></tr>
<tr><td colspan="2">• Hiring manager submits request for new hire within existing budget
• Staffing coordinator defines the parameters of the new position.
• Staffing coordinator publicizes position.
• Applicants submit job applications.
• Staffing coordinator selects shortlisted applicants.
• Hiring manager identifies applicants to interview.</td><td colspan="2">• Staffing coordinator sets up interviews.
• Hiring manager and other interviewers perform interviews.
• Hiring manager and other interviewers provide feedback from the interviews.
• Hiring manager makes hiring decisions.
• Staffing assistant sends offer letters or rejections.
• Successful applicant accepts or rejects job offer or negotiates further.</td></tr>
<tr><th>Participants</th><th colspan="2">Information</th><th>Technologies</th></tr>
<tr><td>• Hiring managers
• Staffing coordinator
• Applicants
• Staffing assistant
• Other interviewers</td><td>• Job requisition
• Job description
• Advertisements
• Job applications
• Cover letters
• Resumes</td><td>• Short list of applicants
• Information and impressions from the interviews
• Job offers
• Rejection letters</td><td>• HR portal for communicating with applicants
• Word processor
• Telephones
• Email</td></tr>
</table>

Work system snapshots increase syntactic expressiveness through a metamodel with seven concepts: *Work System, Customer*, *Product/Service*, *Activity*, *Participant*, *Information*, and *Technology*. Cardinalities in the metamodel express internal consistency rules for the snapshot, e.g., each product/service must be received and used by at least one customer group. The type of one-page tabular representation in Table 2 helps in visualizing a work system's scope by focusing on its core components. There is no need for a modeling procedure for producing this type of table. Note that customers may be work system participants, as in custom software development.

P4. Resources used and produced by specific activities. While a work system snapshot such as Table 2 is useful for discussing a work system's purpose and scope, many stakeholders need a deeper understanding of which resources are used and produced by each activity. Tables in the form of Table 3 are more useful for clarifying operational details by listing selected activities of a work system snapshot along with selected types of resources that are used and/or produced by those activities.

Table 3. Illustration of P4: Selected resources used by a subset of Table 2 activities

Activity	**Actors**	**Information used, created, updated, or deleted**	**Technology**	**Trigger**	**Pre-conditions**	**Post-conditions and product/ services produced**
Submit request for new hire.	Hiring manager	Hiring budget Job requisition	HR portal	Need for new employee	Sufficient hiring budget	Job requisition exists
Define parame-ters of the job.	Staffing coordina-tor	Job requisition Job description Hiring policies	Word processor, HR portal	Job requisi-tion	Job requisition	Job description
Publicize the job opening	Staffing coordina-tor	Experience with advertising media, Advertisement	HR portal, Web site for selected media	Job requisite-ion, Job descript-tion	Job requisition, Job description	Advertisement displayed on websites
Submit applica-tion	Applicant	Job description, Cover letter, Job application, Resume	HR portal	Advertise ment displayed on websites	Advertiseme nt displayed on websites	Receipt of cover letter, job application, and resume

Various metamodels can be the basis of Table 3. A minimalist metamodel for P4 would include *Work System, Activity, and Resource*. It would treat all but the first column in Table 3 as resources that are used by the activities. Saying that actors are "used" by activities may sound strange, but it fits with the way some managers use the term resource in planning and management. For modeling this provides a symmetrical way to handle human, informational, and technological resources. A more expressive metamodel might include all of the column headings in Table 3. An even more expressive metamodel presented in [27] contains over 50 concepts that identify other associations between activities and resources, such as which business rules affect customer-facing activities. Those increasingly elaborate metamodels move toward the level of specificity that programmers need, e.g., in identifying resources that are used or produced. The resulting models are still somewhat informal since the concepts are introduced with natural language.

P5. High precision description of the work system. Existing diagrammatic modeling techniques address many typical needs for understanding how the work system components are structured and how the work system operates. For example, it is easy to represent activity sequence and branching logic using BPMN diagrams with activities in swimlanes for different participant roles (see Figure 3, which also applies to P6). While analysts might prefer a full version of BPMN, business stakeholders might prefer a simpler, restricted version of BPMN based on a minimalist metamodel whose concepts are limited to *Event*, *Activity*, *Gateway*, and *Sequence Flow*. That would suffice for diagramming activities in Tables 2 and 3 even though it would require implicit handling of branching logic for situations such as when an applicant is rejected. Similarly, entity-relationship diagrams and ArchiMate's application and technology layers could be used for P5 level descriptions of data structures and interactions between hardware and software. Thus, a high precision description of the work system needs to use modeling techniques that are not associated with WSM or WST but would be important to include in WSMM if it is to address needs of P5, P6, and P7.

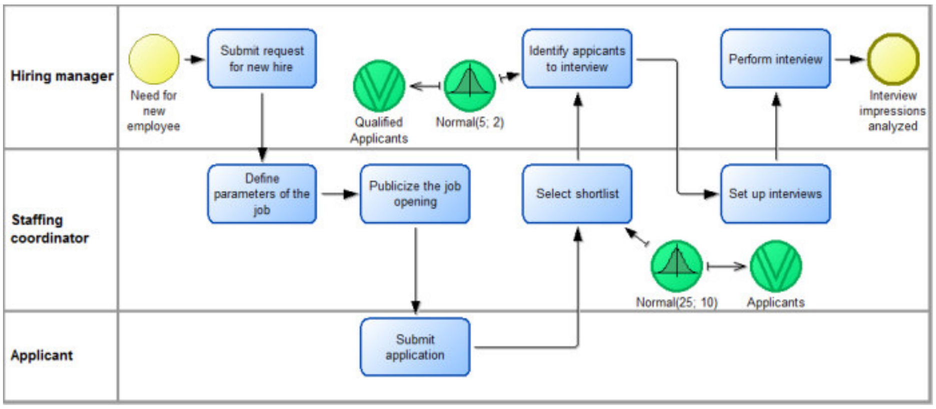

Figure 3. Part of a model supporting a P6 simulation of the hiring process

P6. Simulation of the work system's key processes. Some stakeholders may want to execute simulations to support deeper analysis of the "as-is" work system and deciding on the best of several possible future "to-be" work systems. For example, it might be useful to simulate the workload of different actors depending on the number of applicants, number of interviewers, and other factors. A simulation model could apply an extended BPMN metamodel that adds simulation-specific concepts such as *Statistical Distribution*, and *Random Generators* (the green elements in Figure 3), plus attributes such as transition conditions, probabilities, quantity, cost, and time. Figure 3 illustrates how those concepts might be added to a BPMN model for the hiring example.

P7. Code generation. Figure 2 mentioned code generation because it is a central concern of many researchers in the modeling community. WSMM supports the effort to create understandings and artifacts that are needed in model-driven development but does not attempt to bridge the final gap between understandings and code.

5 Discussion and Conclusions

This paper was inspired by recently published concerns of leading researchers regarding current limitations of enterprise and process modeling. Its goal was to characterize a practical modeling method that applies a central work system metaphor and can be used by stakeholders with different purposes and different levels of technical expertise. WSMM's approach to modeling methods relaxes widely accepted assumptions concerning the nature of modeling methods. That approach positions modeling methods and modeling techniques in a design space that traverses seven types of purposes that had not been articulated in that manner previously and that assumes different levels of technique specificity might be applied for any of those purposes.

We conclude with three topics: whether the four requirements for WSMM were met, how coherence is maintained, and areas for future research.

5.1 Satisfying Four Requirements for WSMM

WSMM satisfies the four requirements identified as the outset. 1) In contrast with most modeling methods, it respects stakeholder diversity by recognizing that different stakeholder purposes may generate different needs for expressiveness, rigor, and completeness. 2) It includes multiple modeling techniques, as shown in the previous section. 3) It can use multiple modeling languages based on different metamodels, but all related to the same work system metaphor. 4) It includes diagrams where needed by stakeholders, e.g., for P5 and P6, but does not require diagrams for other purposes.

Similarly, WSMM addresses other issues mentioned at the outset. WSMM conforms with the willingness in [2] to soften requirements for completeness, coherence, and rigor to achieve broader and more effective usage of modeling. In relation to process modeling problems discussed in [3]: 1) WSMM assumes that different users with different purposes will prefer models with different characteristics; 2) WSMM does not straitjacket processes into one monolithic model; 3) WSMM does not use static decomposition; 4) WSMM does not treat people as machines doing a single task; 5) WSMM is clear about vagueness in its recognition of unstructured and semi-structured processes, adaptations, and workarounds; 6) WSMM is designed to help people improve work systems and does not abstract away from things that matter, such as performance. In relation to issues from [4], WSMM assumes that stakeholders will work at different levels of abstraction and specificity and that their needs may not be satisfied by a specific modeling language.

5.2 Coherence within WSMM

The challenge of coherence in WSMM can be viewed as making sure that WSMM is more than just an assemblage of techniques. At minimum, the use of metamodels that all build on the same core metaphor (the work system) should facilitate production of consistent models for different purposes. For example, a P2 model of capabilities should create a logical constraint on a P3 model that summarizes the work system scope and operation that enacts those capabilities. P3 processes and activities become the basis for the activities in P4 models, and so on.

A larger question is whether WSMM would help business professionals understand work systems for their own purposes while also helping them collaborate with IT professionals with different world views. The use of different metamodels developed for different purposes reduces the likelihood of an automatic way to zoom between the various modeling techniques. However, the progression of the different stakeholder purposes (P1 through P7) outlines a path for communicating between stakeholders who have different purposes. P1 names the work system in the form of the verb phrase. All stakeholders should be able to rally around creating or improving that work system. P2 and P3 support relatively informal understandings of the work system's content, capabilities, and scope. A software engineer in a work system improvement project needs to understand levels P1, P2, or P3 to communicate effectively with business professionals about topics other than isolated details, and also to make it less likely that software will miss basic issues. P4 covers details that business professionals need to verify and that IT professionals need to understand to produce software. Both business and IT professionals might have their own needs for delving into detailed process models, data models, or technical interface models created for P5. In all cases, knowledge of models for P1 through P4 will help in understanding both details and significance of models for P5. Only specialists will pursue P6 and P7, and it would be difficult for them to do that well without understanding models for P3 and P4.

Much of coherence across the various modeling techniques in WSMM needs to be achieved through looking at models and communicating with other stakeholders. From a general standpoint, the different techniques are completely independent and their purposes might not follow the linear and successive nature as proposed for WSMM. Therefore, a technical realization is not always necessary or feasible and needs to be decided individually for each modeling method and its modeling techniques.

5.3 Areas for Future Research

WSMM could help modelers and users apply a range of valuable modeling techniques to situations that seem difficult to address convincingly without relaxing widely held assumptions about the singular nature of modeling methods and modeling languages. Here are some of the research opportunities that represent next steps:

Exploring general implications of the WSMM vision. This paper showed how relaxing common assumptions about modeling methods enables comprehensive visualization of complex systems such as work systems. We do not know where synergies and conflicts between this view and more established views might take us.

Challenging, extending, and/or validating these ideas requires iterations of discussion, feedback, and possibly revision.

Implementation of a widely accessible WSMM toolkit. This paper focused on explaining ideas rather than illustrating technical implementations. In a parallel effort we demonstrated technical feasibility of implementing WSMM by using ADOxx, a widely used metamodeling development and configuration platform for implementing modeling methods that is available through OMiLAB (the Open Models Laboratory), an open community for the conceptualization of modeling methods [28, 29]. ADOxx provides capabilities that can be used to implement all of the metamodels mentioned in this paper plus all of the modeling techniques found in existing Microsoft Word analysis templates used by MBA and EMBA students in recent years. For example, we created a blank work system snapshot in ADOxx and filled in the details from Table 2. ADOxx also can support other work system modeling techniques such as those related to conformance to sociotechnical principles, anticipation of workarounds, customer responsibilities for specific work system activities, and the value of product/services to specific customer groups. Further work and experience with ADOxx is required to identify the most convenient ways to use its modeling capabilities across the many topics in existing WSM templates and other topics based on new metamodels.

Replication for other test cases. Many aspects of this paper, such as the dimensions of the design space in Figure 2, superimposed a rigorous modeling viewpoint on top of useful but less rigorous ideas associated with WSM as it has existed to date. The same type of exercise should be attempted for other sets of ideas, such as general systems theory, sociotechnical theory, actor network theory, soft system methodology, and so on. That would require experts in those areas of theory and practice to identify modeling techniques that are used or could be used, to specify underlying metamodels, and to explore the possibility of producing modeling methods that researchers and practitioners in those areas would find useful.

Further development of the research stream related to WSM, WST, and extensions. The development of WSM started several decades ago with a focus on issues related to P3 and with little or no attempt at rigor other than trying to define terms and encourage organized thinking about work systems that involved IT. The ideas that became WSM and WST evolved gradually. A first book on WSM appeared in 2006; a first version of a work system metamodel in 2010 as an extension of the core ideas; a first article articulating WST appeared in 2013.

The current research extends that stream of research in many directions. It overcomes the limiting assumption that the research was mostly centered around what Figure 2 would call a P3 analysis by business professionals. It eliminates an outdated assumption of a single work system metamodel that needs to be highly detailed (and that until recently was updated incrementally). The new approach is potentially much more valuable because it calls for many alternative metamodels based on the same central metaphor but designed for different purposes.

This paper described metamodels for six of the seven purposes in Figure 2 and noted that different metamodels with different degrees of specificity could be applied for most of those purposes. Length limitations prevented including diagrams of the metamodels. Follow-on research should look at work system-related metamodels in substantial detail, showing them as diagrams, and most importantly, explaining the stakeholder issues that would call for greater syntactic and semantic expressiveness related to specific issues. That research would contrast incremental extensions at each level based on well-known stakeholder concerns (e.g., related to different types of information, product/service offerings, value for customers, application of encapsulated services, and various types of interactions with other work systems) with the combination of more unstructured relationships among modeling techniques.

References

1. Clark, T., Frank, U., Kulkarni, V.: Supporting Organizational Efficiency and Agility: Models, Languages and Software Systems, https://www.dagstuhl.de/de/programm/kalender/semhp/?semnr=16192.
2. Sandkuhl, K., Fill, H.-G., Hoppenbrouwers, S., Krogstie, J., Matthes, F., Opdahl, A., Schwabe, G., Uludag, Ö., Winter, R.: From Expert Discipline to Common Practice: A Vision and Research Agenda for Extending the Reach of Enterprise Modeling. Bus. Inf. Syst. Eng. 60, 69–80 (2018).
3. van der Aalst, W.M.P.: What makes a good process model? Softw. Syst. Model. 11, 557–569 (2012).
4. Karagiannis, D.: Agile modeling method engineering. In: Karanikolas, N.N., Demosthenes, A., Mara, N., Vergados, D., and Michalis, X. (eds.) Proceedings of the 19th Panhellenic Conference on Informatics - PCI '15. pp. 5–10. ACM Press, Athens, Greece (2015).
5. Fettke, P.: How Conceptual Modeling Is Used. Commun. Assoc. Inf. Syst. 25, 571–592 (2009).
6. Mendling, J., Reijers, H.A., van der Aalst, W.M.P.: Seven process modeling guidelines (7PMG). Inf. Softw. Technol. 52, 127–136 (2010).
7. Haisjackl, C., Soffer, P., Lim, S.Y., Weber, B.: How do humans inspect BPMN models: an exploratory study. Softw. Syst. Model. 17, 655–673 (2018).
8. Johannsen, F., Leist, S., Braunnagel, D.: Testing the Impact of Wand and Weber's Decomposition Model on Process Model Understandability. ICIS 2014 Proc. (2014).
9. Mendling, J., Recker, J., Reijers, H.A., Leopold, H.: An Empirical Review of the Connection Between Model Viewer Characteristics and the Comprehension of Conceptual Process Models. Inf. Syst. Front. (2018).
10. Langer, P., Mayerhofer, T., Wimmer, M., Kappel, G.: On the Usage of UML: Initial Results of Analyzing Open UML Models. In: Fill, H.-G., Karagiannis, D., and Reimer, U. (eds.) Modellierung. p. 21 (2014).
11. Muehlen, M. zur, Recker, J.: How Much Language Is Enough? Theoretical and Practical Use of the Business Process Modeling Notation. In: Seminal Contributions to Information Systems Engineering. pp. 429–443. Springer Berlin Heidelberg, Berlin, Heidelberg (2013).
12. Hinkel, G., Kramer, M., Burger, E., Strittmatter, M., Happe, L.: An empirical study on the perception of metamodel quality. In: 2016 4th International Conference on Model-Driven Engineering and Software Development (MODELSWARD). pp. 145–152 (2016).
13. Simões, D., Antunes, P., Carriço, L.: Eliciting and Modeling Business Process Stories. Bus. Inf. Syst. Eng. 60, 115–132 (2018).
14. Sweller, J.: Cognitive load theory, learning difficulty, and instructional design. Learn. Instr. 4, 295–312 (1994).
15. Karagiannis, D., Kühn, H.: Metamodelling Platforms. In: Third International Conference EC-Web 2002. p. 182 (2002).

16. Karagiannis, D., Mayr, H.C., Mylopoulos, J.: Domain- Specific Conceptual Modeling. Springer Berlin Heidelberg (2016).
17. Ferstl, O.K., Sinz, E.J.: Grundlagen der Wirtschaftsinformatik. Oldenbourg, München (2013).
18. Alter, S.L.: How should business professionals analyze information systems for themselves? Presented at the (1995).
19. Alter, S.: The work system method: connecting people, processes, and IT for business results. Work System press, Lankspur, CA, CA (2006).
20. Alter, S.: Work System Theory: Overview of Core Concepts, Extensions, and Challenges for the Future. J. Assoc. Inf. Syst. 14, 72–121 (2013).
21. Truex, D., Alter, S., Long, C.: Systems analysis for everyone else: Empowering business professionals through a systems analysis method that fits their needs. In: ECIS 2010 Proceedings. 4 (2010).
22. Bork, D., Fill, H.-G.: Formal Aspects of Enterprise Modeling Methods: A Comparison Framework. In: Proceedings of the Annual Hawaii International Conference on System Sciences. pp. 3400–3409. IEEE (2014).
23. Bork, D., Alter, S.: Relaxing Modeling Criteria to Produce Genuinely Flexible, Controllable, and Usable Enterprise Modeling Methods. In: Fellmann, M. and Sandkuhl, K. (eds.) 9th International Workshop on Enterprise Modeling and Information Systems Architectures (EMISA) (2018).
24. Checkland, P.B.: Soft systems methodology. Hum. Syst. Manag. 8, 273–289 (1989).
25. Alter, S., Recker, J.: Using a work system perspective to expand BPM research use cases for research. J. Inf. Technol. Theory Appl. 18, 47–71 (2017).
26. Zdravkovic, J., Stirna, J., Kuhr, J.-C., Koç, H.: Requirements Engineering for Capability Driven Development. In: Frank, U., Loucopoulos, P., Pastor, Ó., and Petrounias, I. (eds.) IFIP Working Conference on The Practice of Enterprise Modeling. pp. 193–207. Springer, Berlin, Heidelberg (2014).
27. Alter, S., Bolloju, N.: A Work System Front End for Object-Oriented Analysis and Design. Int. J. Inf. Technol. Syst. Approach. 9, 1–18 (2016).
28. Bork, D., Miron, E.-T.: OMiLAB - An Open Innovation Community for Modeling Method Engineering. In: International Conference of Management and Industrial Engineering (ICMIE) - Management in the Innovation Society. pp. 64–77. Niculescu Publishing, Bucharest, Romania (2017).
29. Bork, D., Buchman, R.A., Karagiannis, D., Lee, M., Miron, E.-T.: An Open Platform for Modeling Method Conceptualization: The OMiLAB Digital Ecosystem. Commun. Assoc. Inf. Syst. (2019).