Requirements Engineering for Sociotechnical Systems that May Include Mixed Initiative Interactions between Humans and Machines

Steven Alter
School of Management
University of San Francisco
San Francisco, USA
alter@usfca.edu

***Abstract*—Mixed initiative interactions between humans and machines pose many challenges for requirements engineering (RE), especially in the broader context of sociotechnical systems (STSs), a topic that has been studied and debated since the 1950s. This paper builds on the work system perspective (WSP), which can be used to describe an STS's operation, structure, purpose, and context in substantial depth regardless of whether some of its subsystems include mixed initiative interactions. It uses the acronym MXN to refer to systems that contain such interactions. This paper explains how aspects of the WSP are useful for identifying requirements for STSs in general and for the much smaller set of STSs that include mixed initiative interactions, a topic emphasized by the CFP of RESOSY 2021, the First International Interdisciplinary Workshop on Requirements Engineering for Sociotechnical Systems.**

Keywords—sociotechnical system, mixed initiative interaction, work system, work system perspective

I. Building on a Traditional View of STS

Practices and thinking associated with STSs have been applied, debated, and expanded since the 1950s. Traditional views of STS involve systems whose human participants use technologies for their activities. A 1977 article in the first volume of *MIS Quarterly* (over four decades ago) viewed an STS as a combination of separable technical and social systems [1]. The title of a 2019 *MIS Quarterly* article [2] referred to the "sociotechnical axis of cohesion of the IS discipline." Those articles use a longstanding concept of STS that is much broader than the MXN subsystems that the RESOSY 2021 CFP denotes as STSs.

Distinguishing between the traditional view of STS and the view of STS in CFP is necessary to avoid confusion in this paper. The CFP states that STSs are "systems that are built to aid humans in specific human tasks" and STSs should be addressed as "mixed initiative systems where the computer or the human can take initiative, monitor events, decide what to do next, and perform tasks." That view of STS echoes Licklider's 1960 vision of "man-computer symbiosis" as "an expected development in cooperative interaction between men and electronic computers. It will involve very close coupling between the human and the electronic members of the partnership." [3] In effect, STSs as characterized by the CFP are a relatively new type of subsystem of STSs that have been imagined for many decades. This paper does not speculate about the source of the CFP's appropriation of the term STS.

Goal and organization. This paper shows how a view of STS based on the WSP illuminates requirements-related issues that likely would be overlooked by focusing tightly on the CFP's assumption that STSs are MXNs. A WSP-centric view of STSs and MXNs provides a richer basis for RE since STSs that contain MXNs also include other human activities.

This paper explains how the WSP provides a basis for RE for an STS and for an MXN that might serve as one of its subsystems. It expands on a brief summary of the WSP by highlighting specific WSP-related topics that are useful for analyzing MXNs, including portrayals and characteristics of WSs and WS elements, performance variables, facets of work, functions performed by subsystems, WS design principles, division of responsibilities for specific activities, interaction patterns, and different degrees of smartness in devices and systems. Failure to consider those topics in RE for an MXN or an STS containing an MXN is analogous to engineering a self-driving car without consider the context of use, e.g., weather, inattentive human drivers, road conditions, obstacles, and interactions with vehicles that may or may not be self-driving.

II. Background about Sociotechnical Systems

The STS movement began in England in the 1950s. The essence of the sociotechnical approach is described as follows by Enid Mumford, a long-term leader in the STS movement (see tribute [4]): "Throughout its history, practitioners have always tried to achieve its two most important values: the need to humanize work through the redesign of jobs and democracy at work. In order to realize these goals, the objective of socio-technical design has always been 'the joint optimization of the social and technical systems'. Human needs must not be forgotten when technical systems are introduced. The social and the technical should, whenever possible, be given equal weight."[5, p. 321] … "The most important thing that socio-technical design can contribute is its value system." … "This tells us that although technology and organizational structures may change, the rights and needs of the employee must be given as high a priority as those of the non-human parts of the system." [5, p. 338]

After summarizing the development and application of sociotechnical thinking, [5] expresses disappointment and doubts about its limited influence in the world of 2006, the year when Mumford died. One explanation is that the underlying ideas of STS have spread to so many different domains that it has become diluted to "a banner under which many different concepts and design principles can flourish that have little relation to one another." [6, p. 234]. For example, [7] identifies four major variants on STS theory and practice: North American STS, Australian STS, Scandinavian STS, and Dutch STS. On the other hand, the diffusion of STS ideas over many decades could be viewed as a success. For example, [8, p. 9] notes that "the work design and processes of both STS and flexible manufacturing have been successfully integrated into most organizations today. It is difficult to find an organization that does not encourage team work, employee participation and decision making" even though "STS began to disappear both academically and in practice in the late 80s early 90s."

Part of the discussion of STS frequently assumes that an STS can be divided into a technical system and a social system (e.g., [1, 5]). That approach has serious shortcomings for RE because the social and technical systems overlap [9]. Many processes in STSs are both social and technical because humans doing some of the work may not conform with specifications due to social issues. The information in an STS is both social and technical because it includes computerized information and interactions between humans. Even technologies often have social aspects since many STS participants use their own computers, smartphones, and other technologies whose selection is partly social in nature.

The WSP addresses that difficulty by treating a WS as a single integrated system, thus eliminating the separation between social and technical systems and covering both STSs with human participants and totally automated systems. Its explicit attention to WS participants and their characteristics and concerns recognizes humanistic values instead of focusing primarily on technical specifications.

III. Summary of the Work System Perspective

The WSP has evolved over many years. Its development started with an attempt to create a systems analysis method for business professionals, which was articulated as the work system method (WSM) [10]. The ideas underlying WSM were formalized as work system theory (WST), and subsequent developments related to service systems, workarounds, design principles, and other topics have been viewed as extensions of WST [11]. The core of WSP is work system theory (WST), which applies equally to WSs in general and to ISs. WST's three components are the definition of WS, the work system framework, and work system life cycle model. Since ideas related to WST and WSM have been presented many times, this section will focus on key points that minimize misunderstanding of the entire approach. The following summary includes a WS interpretation of the idea of STS.

Definition of work. The WSP assumes that work is the application of human, informational, physical, and other resources to produce product/services for internal or external customers (or for oneself). Work can occur in businesses, governments, homes, and other situations where resources are used purposefully to produce outcomes.

Definition of WS. A work system is 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 and/or external customers (or for themselves) [11]. The first *and/or* addresses trends toward automation of work by saying that WSs may be STSs (with human participants doing some of the work) or totally automated. A key point is that many of the same WS ideas apply equally to sociotechnical WSs and totally automated WSs and to MXNs. Those ideas include many of the properties of the elements shown in Figure 1.

Special cases. The most important distinction in describing special cases of WS is the difference between a sociotechnical WS in which human participants perform some of the activities vs. totally automated WS where all activities are performed by machines. That distinction says that a MXN is a type of STS regardless of whether it is an IS or is a system devoted to physical activities.

An IS is a WS most of whose activities are devoted to capturing, transmitting, storing, retrieving, deleting, manipulating, and/or displaying information. This definition differs from 20 previous definitions in [12] and was one of 34 definitions of IS noted in [13]. It differs from assuming an IS is a tool that is "used" or that an IS exists to produce representations of real world systems [14]. An example is a sociotechnical accounting IS in which accountants decide how specific transactions and assets will be handled for tax purposes and then produce monthly or yearend financial statements. This is an IS because its activities are devoted to processing information. It is also supported by a totally automated IS that performs calculations and generates reports. In both cases, an IS that is an integral part of another WS cannot be analyzed, designed, or improved thoughtfully without considering how IS changes affect that WS. The same idea applies to MXNs that are subsystems of larger WSs.

Projects, service systems, self-service systems, and some supply chains (interorganizational WSs) are other important special cases. For example, software development projects and other projects are WSs designed to produce specific product/services and then go out of existence. Thus, a project that creates or improves a MXN is a WS on its own right.

Consistent with other ideas in the WSP, MXNs can be viewed as a highly restricted special case of WSs. The fact that a WS contains a MXN subsystem does not imply that the WS itself should be viewed as a MXN (in the same way that a WS that uses IT typically should not be viewed as an IT system). MXNs may not be ISs because the humans and/or totally automated parts of an MXN may perform physical work.

Work system framework: a basic understanding of a WS. The nine elements of the WS framework (Fig. 1) are the elements of a basic understanding of a WS's form, function, and environment during a period when it is stable enough to retain its identity even though incremental changes may occur, such as minor personnel substitutions or technology upgrades. *Processes and activities*, *participants*, *information*, and *technologies* are completely within the WS. *Customers* and *product/services* may be partially inside and partially outside because customers often participate in activities within a WS and because product/services take shape within a WS. *Environment*, *infrastructure*, and *strategies* are outside of the WS even though they have direct effects within a WS and may be affected by major changes in significant WSs.

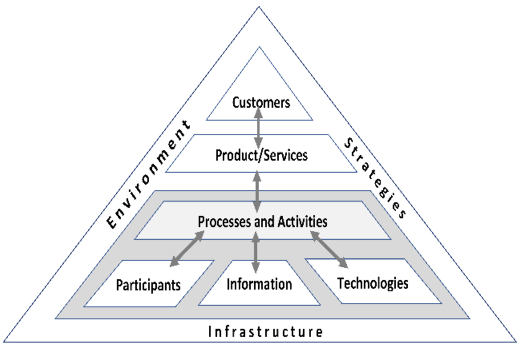

Fig. 1. The work system framework

The following clarifications are often useful: *Customers* refers to people or organizations that receive product/services produced by a WS. This includes internal and external customers. The term *product/services* is used to bypass controversies about special characteristics of products vs. services. The term *processes and activities* is used because the activities in some WSs are not structured as processes. *Infrastructure* refers to human, informational, and technical resources that are viewed as shared by multiple WSs instead of being associated primarily with one WS. An example of human infrastructure is an IT group that can be viewed as a resource used by multiple WSs. "Elements of the WS framework" will be abbreviated as "WS elements" even though the last three elements are viewed as outside of a WS and often are controlled elsewhere.

Work system life cycle model (WSLC): how WSs change over time. ISs and other WSs evolve through a combination of planned change through projects and unplanned change through adaptations and workarounds (Fig. 2). WSLC phases (initiation, development, implementation, operation and maintenance) may be performed in different ways. Typical activities and responsibilities (e.g., designing, debugging, training, etc.) associated with specific phases apply for waterfall, agile, prototyping, use of off-the-shelf applications, and shadow IT, even when several phases overlap or iterate.

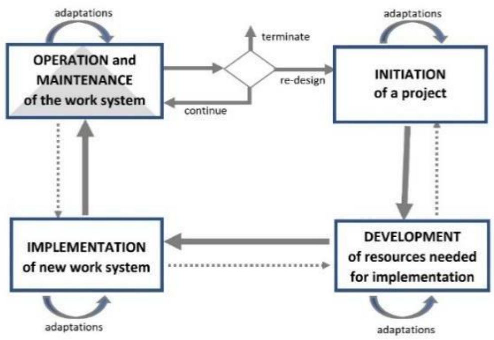

Fig. 2. The work system life cycle model (WSLC)

Both planned and unplanned changes often affect multiple WS elements, not just technologies. The *development* phase creates or acquires and then tests software and other resources needed for implementation in the organization. The *implementation* phase involves much more than installation of software on computers. The WSLC's four idealized phases (and related sub-phases) express a waterfall-like approach to identifying things that should happen as a WS evolves iteratively. Many WSLC topics remain valid when agile approaches are used for developing software, such as the importance of WS changes rather than just software development, evolution over time rather than one-time projects, the simultaneous importance of planned and unplanned change, and the relevance of key activities and responsibilities within each phase. The key activities and responsibilities remain even if the phases are partially merged and regardless of whether the WS uses homegrown software, commercial application software, or external platforms. For example, regardless of whether aspects of development and implementation are partly merged, it is still necessary to determine requirements (at an appropriate level of detail), acquire, produce, or fix software that is needed, test and debug software, decide how to implement WS changes, identify implementation problems, train WS participants, and so on.

Coverage of sociotechnical systems. The WSP covers all operational systems in organizations, including STSs that can be viewed as WSs and totally automated systems that can be viewed as WSs. Those systems include all ISs, projects, and other special cases of WS.

Thus, the WSP covers a much broader domain than the domain identified in the CFP for RESOSY2021. The CFP prioritizes "systems built to aid humans in specific human tasks" [in situations that involve] "mixed initiative … where the computer or the human can take initiative, monitor events, decide what to do next, and perform tasks." The WSP covers those situations and many others. It covers "systems built to aid humans in specific human tasks" but it also covers systems that use automation to replace people who previously performed specific tasks and also systems that perform totally automated tasks that were never performed by people. The WSP covers systems with mixed initiative interactions of humans and computers, but also covers systems where responsibilities of humans and computers are structured to be completely separate.

A key issue that bears reiteration is the interpretation of the terms *system* and *STS* in regard to this discussion. Computer science papers often assume that systems are software-controlled entities that operate on computers. In contrast, the WSP covers both sociotechnical WSs (where some of the work is done by humans) and totally automated WSs (where all of the work is automated). Many computer science techniques that focus specifically on software are more effective than the WSP for understanding nuances of software and software development. This paper's emphasis lies elsewhere, i.e., in explaining how the WSP provides insights for RE for STSs in general and for MXN subsystems of STSs.

IV. WSP VIEW OF REQUIREMENTS ENGINEERING

This section uses ideas from a 2021 ACM Tech Talk [15] by Bertrand Meyer to summarize the nature of requirements engineering (RE) and to illustrate a WSP view of RE. The next section will focus on aspects of the WSP that are especially relevant to RE for STSs that contain subsystems "built to aid humans in specific human tasks" and that involve "mixed initiative … where the computer or the human can take initiative, monitor events, decide what to do next, and perform tasks." (the domain described by the CFP).

According to [15], RE can be described in terms of PEGS (Project, Environment, System, and Goals) that are equally applicable to waterfall projects and to agile projects that proceed through iterations of sprints. Each element of PEGS is reflected in the WSP. In the WSLC, formal *projects* occur through initiation, development, and implementation phases each of which involves activities mentioned earlier. RE occurs during the initiation phase and may continue during the development phase. *Goals* are set during the initiation phase and may be revised in subsequent phases. The surrounding *environment* appears as one of nine WS elements (Fig. 1) and is reflected in the deliberations during the initiation phase of the WSLC. The *system* is a WS that may have multiple sociotechnical or totally automated subsystems, each of which can be analyzed or designed based on WST and WSM.

Below are brief descriptions of four WSs that could involve mixed initiative interactions between a person and an automated entity that will be called a robot even though it may or may not have a physical realization (e.g., as in robotic process automation). The sketches distinguish briefly between the main WS and an MXN. In all four cases, a realistic RE effort would need to cover the WS (the *system* in PEGS terms) within which the MXN exists as a subsystem. Failure to consider the larger WS would result in requirements that treat the MXN's environment too narrowly to permit evaluation of its effectiveness for meeting WS *goals*.

An interactive tutor. The broad class of WSs that provide instruction may or may not include an MXN whose activities are controlled partly by a student and partly by a robot serving as an automated tutor. The extent of shared control can be described along a dimension that starts with the student merely answering questions from the robot. Highly interactive learning is more like a dialogue between the student and the robot. That dialogue is part of a larger WS that may involve other activities such as recording the student's progress and understanding of specific items in the material to be learned, assignment of students to learning programs, monitoring the continuity of the student's attention, and so on.

A customer service chatbot. The chatbot is part of a larger WS of providing customer service. For example, a presentation [16] about the Moveworks capability for IT help desks noted that its chatbot answers around 40% of IT help desk queries and escalates the rest to human customer service representatives who may escalate queries further if needed. The 40% reduction in queries handled by humans reduces customer service costs and eliminates many delays. The extent to which the chatbot is an MXN was not clear from [16].

An imagined flight control system. The WS in this instance can be summarized as flying a small personal aircraft from a starting point to a destination. Activities in that WS include deciding on the flight plan, performing a safety check, taking off, flying to the destination, landing, and performing aircraft-related activities that occur after landing. A robotic component of an MXN could inform the pilot about problems related to weather along the flight plan. The pilot could ask the robot to estimate how much fuel will remain when the aircraft reaches its destination. Also, the pilot could ask the robot to suggest a modification of the flight plan if unexpected turbulence occurs. In effect, the robot would offload some of the activities normally performed by pilots and, perhaps, air traffic controllers.

An imagined RE assistant. RE can be viewed as a type of WS in which analysts perform activities directed at producing requirements. Applying the definition of WS, RE is a system (a WS in its own right) in which human participants *and/or* machines perform processes and activities using information, technology, and other resources to produce specific product/services for internal and/or external customers. By that definition, some future type of RE might be partly or totally automated. The requirements are product/services that are produced, and the customers are people or organizations that receive and use the requirements.

Assume (quite optimistically) that knowledge about RE is sufficiently codified that an MXN subsystem of a WS for RE could include a robot that monitors the current state of a structured requirements document. The robot could ask human analysts questions such as whether noncompliance via human agency has been considered or whether past workarounds have indicated areas where the target WS needs to be improved. Conversely, the analyst could ask the robot to examine the current state of the requirements document and to apply codified knowledge about specific aspects of RE to identify issues such as whether serious conflicts exist between different user stories.

V. Aspects of the Work System Perspective That Are Pertinent to Requirements Engineering for MXNs

Topics within the WSP build outward from the definition of WS and include the main ideas in WST, the application of those ideas in WSM, and a series of extensions related to design principles, service systems (in a business sense), workarounds, WS interactions, and other topics. Covering the entire WSP would require a book-length discussion. Since that is not practical for current purposes, this section emphasizes WSP topics that are relevant to all WSs but are especially relevant to MXNs. Those topics include portrayals and characteristics of WSs and WS elements, performance variables for WSs and WS elements, facets of WSs and WS elements, functions performed by WS subsystems, WS design principles, division of responsibilities for specific activities, interaction patterns, and different degrees of smartness in devices and systems.

A. *Portrayals of WSs and WS elements*

Portrayals of WSs are alternative concepts for visualizing the entirety of a WS or WS element. (See Fig. 3.) The following list identifies alternative portrayals that are relevant to many WSs. For each portrayal, the list includes a question or issue that could be relevant to RE for a specific MXN.

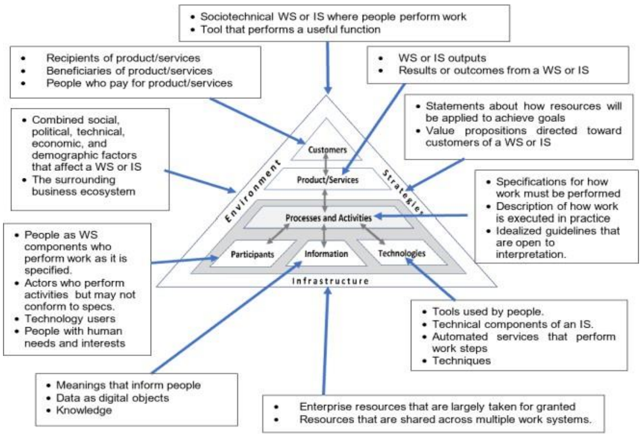

Fig. 3. Portrayals of WSs and WS elements

- Customers might be portrayed as recipients of product/services or as beneficiaries of product/services. Does this MXN have any customers by either portrayal?
- Product/services might be portrayed as outputs that are delivered vs. as results of extensive collaboration. What identifiable product/services does this MXN produce?
- Processes might be portrayed as idealizations of how work should be done vs. as descriptions of how work is executed. How does RE for this MXN consider the possibility of noncompliance with specifications?
- Participants might be portrayed as WS components that follow specifications or as people with human needs and interests. To what extent are both portrayals used in RE for this MXN?

- Information might be portrayed as knowledge, as a conveyor of meanings that inform people, or as machine-processed digital objects. How does RE for this MXN apply those different views of information?
- Technologies might be portrayed as tools used by users who perform work vs. as technical components of a WS vs. as automated services that perform work. How does RE for this MXN reflect those portrayals?

B. Characteristics of WSs and WS elements

In the WSP, characteristics are properties used for describing or analyzing WSs, WS elements, or other resources. As shown in Fig. 4, characteristics of a WS as a whole include scalability, flexibility, resilience, degree of centralization, and fragility. Characteristics of processes and activities include degree of structure, complexity, integration, and rhythm. Characteristics of information include precision, age, traceability, usability, and bias.

Key characteristics for WSs as a whole (the top of Fig. 4) are also important for MXNs, especially where RE issues related to the level of scalability, flexibility, resilience, capacity, and agility for an entire WS may have direct impacts on RE for a MXN within the WS.

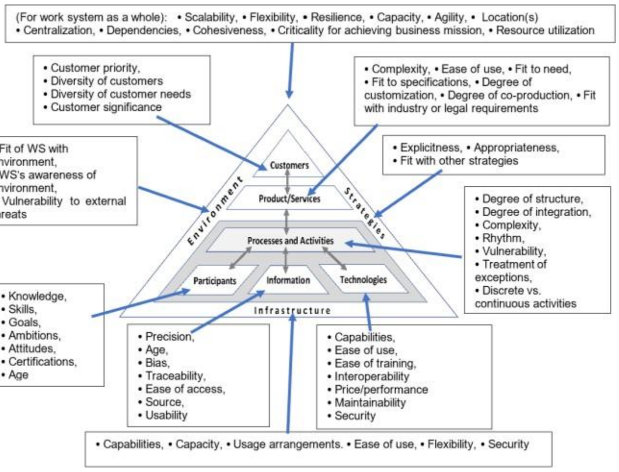

Fig. 4. Characteristics of WSs and WS elements

Other characteristics in Fig. 4 also have direct implications for MXNs. A high degree of structure in a MXN's processes and activities implies that interactions between humans and machines are largely about following scripts, whereas a less structured MXN would allow much less scripted interactions that could require a semblance of smartness or intelligence [17]. Information also brings interesting RE questions for MXNs, such as how to describe or limit information-related bias on the part of the human participant or the robot. Realistic RE analysis should say something about the knowledge and skill that MXN participants need to bring to interactions within the MXN. Assumptions about a participant's personal goals and ambitions should be included in RE because playing a role in a MXN might or might not be consistent with those goals and ambitions.

C. Performance variables for WSs and WS elements

Performance variables in the WSP (Fig. 5) are concepts used for describing or analyzing how well entities or their constituents operate. Required levels of performance variables would be viewed as "non-functional requirements" in the world of software.

Errors and delays in other parts of the WS that an MXN serves will likely affect the operation of the MXN. Thus, metrics related to a MXN's performance at specific times often might depend on the state and operation of other parts of the WS in which the MXN exists. For example, downtime or errors in processes that provide inputs to the MXN could cause the MXN to operate more slowly or to stop operating at all, which likely would affect the metrics for the MXN and its human participants during that period. Other important performance variables that might be overlooked during RE involve job performance and job satisfaction of participants in an MXN. For example, participating in the MXN could lead to better or worse job performance and job satisfaction.

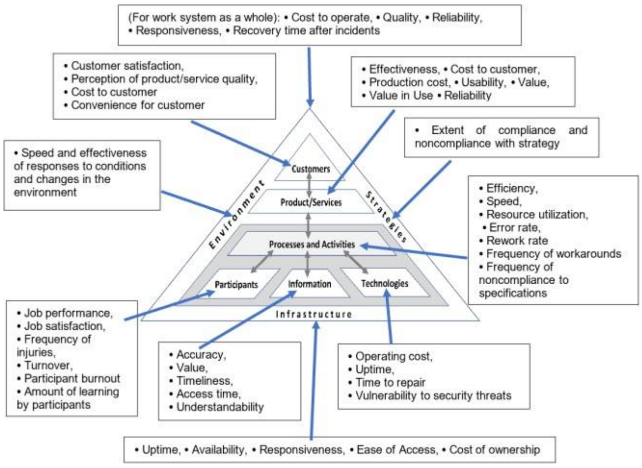

Fig. 5. Performance variables forf WSs and WS elements

D. Facets of WSs and WS elements

Facets of entities are alternative faces or aspects of an entity that can be observed or analyzed. The idea of "facet" is like a facet of a cut diamond. It is not a separate component of the diamond, but rather a face or aspect that can be observed or analyzed. Fig. 6 identifies facets of WSs as a whole and of each WS element.

The most useful set of facets for MXN-related RE is the 18 "facets of work" that can be viewed as facets of the processes and activities in a WS (see Fig. 6). Those facets apply to both sociotechnical and totally automated systems, are associated with specific concepts, brings evaluation criteria and design trade-offs, have sub-facets, and bring open-ended questions for starting conversations [18]. Some facets overlap (e.g., making decisions and communication). Whether or not to include a concept as a facet of work was based on that concept's association with useful concepts, evaluation criteria, and design trade-offs. For example, making decisions, communicating, and providing information all are associated with useful concepts, evaluation criteria, and design trade-offs. The 18 facets were the end-product of an iterative design process that might have led to 14 or 27 facets. Future research might lead to a different set of facets of work.

The central contribution of facets of work for RE related to MXNs is that the facets of work provide a way to be specific about requirements for many specific types of capabilities that otherwise might have been overlooked. For example, consider the facets *learning*, *planning*, *improvising*, and *maintaining security*. Having a list of facets makes it less likely that those topics will be overlooked in RE related to MXNs and to WSs in general. Linkage of each element of that list to some version of associated concepts, evaluation criteria, design trade-offs, sub-facets, and open-ended questions identified in [18] could provide further support for RE.

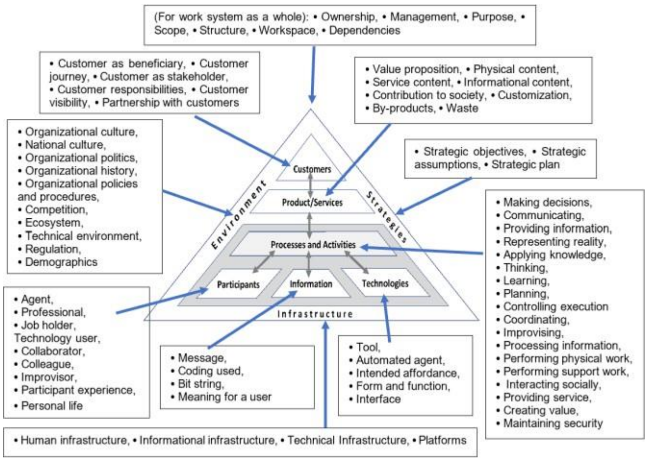

Fig. 6. Facets of WSs and WS elements

E. Functions performed by subsystems

The following list identifies a variety of functions that might be performed through interactions between a human participant and a robot within an MXN. This list was first imagined in relation to functions that an IS might perform to support a WS, with the assumption that the list might be expanded through a structured analysis of IS case studies.

- providing access to information,
- defining and enforcing rules for collecting or sharing information,
- providing methods for aggregating information,
- providing methods for analyzing information,
- controlling activity sequences in workflows,
- enforcing compliance with business rules,
- creating alarms when predefined conditions occur,
- controlling or facilitating coordination,
- suggesting decisions,
- triggering automated functions,
- performing totally automated tasks autonomously.

This list shows that RE for an MXN (or other WS) could be supported by a list of common functions even though it says nothing about whether the person performs those functions for the robot or vice versa. Many other functions might be included. As with the facets of work, this type of list could help analysts make sure they have considered a range of common possibilities. More broadly, some version of a list of functions potentially helps in realizing that RE for a MXN should be specific about functions being performed regardless of whether they are initiated by either a human or a robot.

F. WS design principles

Design principles are statements that express desired properties of designed entities within a domain. Design principles may apply to all WSs within a domain, to specific types of WS within the domain, and/or to WSs associated with a community of practice.

Fig. 7 uses the format of a "work system snapshot," a basic tool from the WSM, to organize 24 design principles related to sociotechnical WSs. Each design principle could be stated more elaborately, more like a fully specified software design pattern that is viewed as a reusable solution to a commonly occurring problem (e.g., [19]). Unlike axioms or laws, design principles often have exceptions, may be mutually inconsistent, and may conflict in practice. For example, as noted in [20], in some cases the principle "please the customers" may conflict with "do the work efficiently."

Many of the design principles in Fig. 7 (or other design principles that have been proposed) could be applied during RE for MXNs. For example, the design principles in the part of Fig. 7 for processes and activities includes design principles related to variability, efficiency, judgment, problem control, quality control, boundaries between steps, and the match between work practices and participants. All of those ideas could be explored as part of an RE effort focused on an MXN.

Customers		Product/Services
• #1: Please the customers. • #2: Balance priorities of different customers.		
Processes and Activities		
• #3: Match process flexibility with product variability • #4: Perform the work efficiently. • #5: Encourage appropriate use of judgment. • #6: Control problems at their source. • #7: Monitor the quality and timing of both inputs and outputs. • #8: Boundaries between steps should facilitate control. • #9: Match the work practices with the participants.		
Participants	Information	Technologies
• #10: Serve the participants. • #11: Align participant incentives with system goals. • #12: Operate with clear roles and responsibilities.	• #13: Provide information where it will affect action. • #14: Protect information from inappropriate use.	• #15. Use cost/effective technology. • #16: Minimize effort consumed by technology.
Infrastructure	• #17: Take full advantage of infrastructure.	
Environment	• #18: Minimize unnecessary conflict with the external environment	
Strategies	• #19: Support the firm's strategy	
Work System as a Whole	• #20: Maintain compatibility and coordination with other work systems. • #21: Incorporate goals, measurement, evaluation, and feedback. • #22: Minimize unnecessary risks. • #23: Maintain balance between work system elements. • #24: Maintain the ability to adapt, change, and grow.	

Fig. 7. Design principles for sociotechnical WSs

G. Division of responsiblities for specific activities

An important design question for MXNs is the division of responsibilities, i.e., the extent to which the person or the machine is responsible for each activity in a MXN subsystem. RE for a MXN would be superficial if it did not deal with that question either in a general way or by being explicit about whether a human or a machine is responsible for initiating each activity, for monitoring each activity, for declaring an activity complete, and for transitioning to other activities.

A WSM tool called a service responsibility table (SRT) [21] was designed for other purposes but can be used for describing the division of responsibilities in a MXN. An SRT applied to a MXN would be a table consisting of at least three columns: 1) a list of activities in the MXN, 2) responsibilities of a person regarding each of those activities, 3) related responsibilities of an automated entity regarding each of those activities. Additional columns could clarify responsibilities for specific aspects of each activity, such as initiation, quality control, error detection, and declaration of completion. An SRT can also be expanded by adding columns related to topics such as mutual visibility or at least awareness of non-interactive activities performed by the human or machine.

H. WS interactions and interaction patterns

Interactions between WSs include unidirectional, mutual, or reciprocal actions, effects, relationships, influences, or interplay between two or more WSs. Systems theorists such as Ackoff [22] and Checkland [23] observe that systems typically exist to serve other systems and that understanding or analyzing a system requires understanding whatever systems are being served and how those systems are being served. A thorough analysis needs to go further by considering planned and unplanned interactions with other systems regardless of whether they serve or are served by a focal system of primary interest. The many types of interactions between systems range from repetitive interactions such as supplier-customer transactions to transient interactions related to mishaps or malicious actions. A thorough understanding of system interactions should include indirect impacts such as effects of inconsistent goals, inconsistent standards, and inconsistent treatment of personnel. It also should consider direct and indirect impacts when other entities perform unexpectedly or inadequately. In general, RE should consider system interactions both while also focusing on systems in isolation and while focusing on the surrounding context. Thus, even a superficial look at an MXN in the context of RE should consider its interactions with other WSs, with resources used, or with other aspects of the WS in which it operates.

The idea of interaction patterns can be used when thinking about the requirements for capabilities within an MXN. Preliminary research [24] identified 19 interaction patterns within four categories. Those interaction patterns include:

One-way patterns are unidirectional interactions that have been studied in relation to the language action perspective (LAP). Patterns within this category are *inform, command, request, commit,* and *refuse*. All of those patterns involve unidirectional interactions.

Coproduction patterns are bilateral patterns involving jointly produced interactions whose instantiations can be observed as sequences of unidirectional interactions, some of which may be described as speech acts. Coproduction patterns include *converse, negotiate, mediate, share resource*, and *supply resource*. The first three are fundamentally about bilateral speech situations, whereas the other two are fundamentally about coordination as described by coordination theory [25, 26].

Access and visibility patterns are unidirectional patterns concerning one entity obtaining access or visibility related to another entity and about countermeasures to prevent access and visibility. These patterns include *monitor, hide, protect,* and *attack*. The first of these involves a typical management activity. The next two involve defensive maneuvers. The last pattern represents a threat.

Unintentional impact patterns are the least articulated patterns because of the great uncertainty about the sources and effects of many unintentional impacts. Examples include *overlap*, *market-based*, *spillover*, *indirect*, and *accidental* interactions. While it may not be possible to anticipate those impacts, ignoring the possibility that they will occur is certainly not a beneficial RE practice.

While the ideas in [24] surely could be elaborated further, it is worth noting that likely elements of typical interaction patterns in the first three categories include actor roles (e.g., requestor/respondent, initiator/recipient, partner, or intermediary), actor type (e.g., person or machine), actor rights for each role, actor responsibilities for each role, cause or trigger of the interaction, desired outcome, generic process or activities, possible states of an interaction, and alternative enactments. Occasionally relevant elements of interaction patterns include constraints, risks and risk factors, relevant concepts, interaction verification, and interaction evaluation

I. Smartness of devices and systems.

Finally, RE related to MXNs might consider ideas about the smartness of devices and systems since the notion of MXN tends to imply some degree of smartness in a computerized device. An approach to smartness of devices and systems is explained in [17], which identifies generic capabilities that might be executed by computerized algorithms. RE might apply that idea without getting entwined in debates about the definition, nature, or limitations of artificial intelligence. [17] uses four categories to organize numerous capabilities that might be built into devices or systems:

- **Information processing**. Capture information, transmit information, store information, retrieve information, delete information, manipulate information, display information.
- **Action in the world**. Sensing, actuation, coordination, communication, control, physical action.
- **Internal regulation**. Self-detection, self-monitoring, self-diagnosis, self-correction, self-organization.
- **Knowledge acquisition**. Sensing or discovering, classifying, compiling, inferring or extrapolating from examples, inferring or extrapolating from abstractions, testing and evaluating.

As noted in [17], the smartness built into a device or system (in a MXN) for any of the above capabilities can be characterized along the following dimension:

- **Not smart at all**. Does not perform activities that exhibit the capability.
- **Scripted execution**. Performs capability-related activities according to prespecified instructions.
- **Formulaic adaptation**. Adaptation of capability-related activities based on prespecified inputs or conditions.
- **Creative adaptation**. Adaptation of capability-related activities based on unscripted or partially scripted analysis of relevant information or conditions.
- **Unscripted or partially scripted invention**. Invention of capability-related activities using unscripted or partially scripted execution of a workaround or new method.

VI. Discussion and Conclusion

This paper started by noting that the CFP of RESOSY 2021 characterizes STSs as “systems that are built to aid humans in specific human tasks” that should be addressed as “mixed initiative systems where the computer or the human can take initiative, monitor events, decide what to do next, and perform tasks.” That characterization is much more restricted than typical characterizations of STS that researchers and practitioners have used for decades. The acronym MXN (mixed initiative system) was used to pursue the CFP's focus without causing confusion about different views of STS.

This paper's contribution used aspects of the WSP to identify many topics that should be considered in RE related to WSs in general and MXNs in particular. It summarized the main ideas in the WSP and then focused on aspects of the WSP that are especially relevant to RE for MXNs. Those aspects of the WSP include portrayals and characteristics of WSs and WS elements, performance variables, facets of work, functions performed by subsystems, WS design principles, division of responsibilities, interaction patterns, and the characterization of smartness in devices and systems.

This paper illustrated the idea of MXNs by using four examples, but did not propose MXN requirements for those examples. Doing so would have required a detailed discussion of those situations including description of the WSs in which the MXN examples exist.

One of this paper's important limitations is its focus on a specific set of ideas related to the WSP, aspects of which have been presented in many papers on a range of topics. This paper's highly condensed presentation of many ideas outlines an integrated approach to addressing many RE issues related to STSs and MXNs, but it also leaves many questions that can only be answered through much more extensive discussion of specific WSP concepts and assumptions underlying the WSP.

This paper necessarily omitted many ideas that could not fit in a short paper. Other researchers surely would approach topics related to MXNs from other viewpoints such as agentic artifacts [27] or multi-agent systems. Similarly, important issues related to human autonomy and dignity at work [28] could be approached from many other directions. Even topics such as intrusive monitoring at work and the "uncanny valley" [29] (where attempts at social behavior by robots that lack human emotions seem unnatural and untrustworthy) could have been discussed in a much longer paper or even a book.

The next step in extending this paper's ideas is to validate their usefulness by applying those ideas to specific RE projects that focus on WSs that contain MXNs or might contain MXNs in the future. Careful evaluation of documents produced in those projects plus interviews of project participants would probably help in describing the extent to which specific WSP ideas in this paper provide insights or simply seem consistent or inconsistent with existing practices from previous projects.

Mixed initiative systems have many potential applications that go far beyond current practice. Research about whether existing RE techniques need to be extended for such situations is potentially quite valuable. The WSP-centric ideas presented in this paper could be a step forward in that research.

REFERENCES

[1] R.P. Bostrom and J. S. Heinen, "MIS problems and failures: a socio-technical perspective, part II: the application of socio-technical theory," MIS Q., vol. 1, no. 1, 1977, pp.11-28,

[2] S. Sarker, S. Chatterjee, X. Xiao, X. and A. Elbanna, The sociotechnical axis of cohesion for the IS discipline: Its historical legacy and its continued relevance. MIS Q., Vol. 43, No. 3, 2019, pp.695-720.

[3] J.C. Licklider, "Man-computer symbiosis." IRE transactions on human factors in electronics, (1), 1960, pp.4-11.

[4] D. Avison, D., N. Bjørn-Andersen, et al. (19 co-authors)., Enid Mumford: a tribute. Inf. Syst. J.,, Vol. 16, No. 4, 2006. pp.343-382.

[5] E. Mumford, "The story of socio-technical design: Reflections on its successes, failures and potential," Inf. Syst. J., vol. 16, no. 4, 2006, pp. 317–342,

[6] K. Eason, "Afterword: The past, present and future of sociotechnical systems theory," Appl. Ergon., vol. 45, 2013, pp. 1–8,

[7] F. M. v Eijnatten, A. B. Shani, and M. M. Leary, "Sociotechnical systems: designing and managing sustainable organization"s, Handbook of organization development, T.G. Cummings, (Ed.) Sage; 2008.

[8] S. Winby, "The adaptive work system: a perspective on the evolution of socio-technical systems," Socio-Technical Roundtable annual conference. New Orleans, LA, 2011

[9] S. Alter, "Applying socio-technical thinking in the competitive, agile, lean, data-driven world of knowledge work and smart, service-oriented, customer-centric value creation ecosystems." Comp. Syst. Infor. and Mod. Q., Vol. 18, 2019, pp. 1-22,

[10] S. Alter, The work system method: connecting people, processes, and IT for business results. Larkspur, CA: Work System Press, 2006.

[11] S. Alter, "Work system theory: overview of core concepts , extensions, and challenges for the future," J. Assoc. Inf. Syst., vol. 14, no. 2, 2013, pp. 72-121.

[12] S. Alter, "Defining information systems as work systems: implications for the IS field," Eur. J. Inf. Syst, Vol. 17, No. 5, 2008, pp. 448-469

[13] S.K. Boell and D. Cecez-Kecmanovic, "What is an information system?" Proceedings of Hawaii International Conference on System Sciences, HICSS, 2015, pp. 4959-4968, IEEE

[14] A. Burton-Jones, J. Recker, M. Indulska, P. Green, and R. Weber. "Assessing representation theory with a framework for pursuing success and failure." MIS Q., Vol. 41, No. 4, 2017, pp.1307-1333.

[15] B. Meyer, "The four PEGS of requirements engineering," presentation slides, ACM Tech Talk, 4 March 2021. Accessed on 1 Sept. 2021 at https://www.youtube.com/watch?v=JaOcWrV75pE

[16] B. Shah, "How conversational AI will empower one billion knowledge workers, MIT AI conference 2020: AI for a better world, https://www.youtube.com/watch? v=bzBZA_2kkNE, last accessed 28 August 2021.

[17] S. Alter, "Understanding artificial intelligence in the context of usage: Contributions and smartness of algorithmic capabilities in work systems." Int. Journal of Inf. Mgt., published online, August 2021.

[18] S. Alter, "Facets of work: Enriching the description, analysis, design, and evaluation of systems in organizations," Communications of the Association for Information Systems (forthcoming), In Press, 2021.

[19] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, Design Patterns:Elements of Reusable Object-Oriented Software, Reading, MA: Addison-Wesley. 1995

[20] S. Alter and R. Wright, "Validating work system principles for use in systems analysis and design, Proceedings of the International Conference on Information Systems, 2010.

[21] X. Tan, S. Alter, and K. Siau. "Using service responsibility tables to supplement UML in analyzing e-service systems." Dec.Sup.Syst,, Vol. 51, no. 3, 2011, pp. 350-360.

[22] R. L. Ackoff, "Towards a system of systems concepts," Mgt. Sci., Vol. 17, No. 11, 1971, pp. 661-671.

[23] P. Checkland, Systems thinking, systems practice, Chichester, UK: John Wiley, 1999.

[24] S. Alter, "System Interaction Patterns," IEEE Conference on Business Informatics, Paris, France, Aug. 2016.

[25] T. W. Malone and K. Crowston. "The interdisciplinary study of coordination." ACM Comp. Surv., Vol. 26, No. 1, 1994. pp. 87-119,

[26] K. Crowston, J. Rubleske, and J. Howison, "Coordination Theory: A Ten-Year Retrospective". in Zhang, P. and Galletta, D. (Eds.) Human-Computer Interaction in Management Information Systems, Vol I. M. E. Sharpe, 2006.

[27] A. Baird and L. M. Maruping. "The Next Generation of Research on IS Use: A Theoretical Framework of Delegation to and from Agentic IS Artifacts." MIS Q. Vol. 45, no. 1,2021, pp. 315-341.

[28] D. E. Leidner and O. Tona. "The CARE Theory of Dignity Amid Personal Data Digitalization." MIS Q., Vol. 45, no. 1, 2021, pp. 343-370.

[29] M. Mori, K. F. MacDorman, and N. Kageki. "The uncanny valley [from the field]." IEEE Rob. & Autom. Mag. Vol. 19, no. 2, 2012, pp. 98-100.