



Can Information System Engineering Make Cyber-Human Systems Smarter?

Steven Alter^(✉) 

University of San Francisco, 2130 Fulton Street, San Francisco 94117, USA
alter@usfca.edu

Abstract. The term smart is often used carelessly in relation to systems, devices, and other entities such as cities that capture or otherwise process or use information. This exploratory paper takes the idea of smartness seriously as a way to reveal basic issues related to IS engineering and its possibilities and limitations. This paper defines work system, cyber-human system, digital agent, smartness of systems and devices, and IS engineering. It links those ideas to IS engineering challenges related to cyber-human systems. Those challenges call for applying ideas that are not applied often in IS engineering, such as facets of work, roles and responsibilities of digital agents, patterns of interaction between people and digital agents, knowledge objects, and a range of criteria for evaluating cyber-human systems and digital agents. In combination, those ideas point to new possibilities for expanding IS engineering to reflect emerging challenges related to making cyber-human systems smarter.

Keywords: Cyber-Human System · Work System · Information System Engineering · Smartness of Systems and Devices · Facets of Work

1 Steps Toward IS Engineering for Cyber-Human Systems

One might wonder how cyber-human systems are fundamentally different from the vast number of IT-enabled work systems that operate in the world of commerce. That seemingly innocent question leads to many important issues related to describing, analyzing, designing, and evaluating cyber-human systems (and other IT-enabled work systems). The important question is not about fundamental differences between cyber-human systems and other systems. Rather, it is about describing and analyzing cyber-human systems while attending to roles of digital agents within those systems and interactions between digital agents and their human participants.

This paper treats *cyber-human system* (CHS) as a type of work system in which human participants and digital agents interact in nontrivial ways that involve conscious attention by human participants. This paper summarizes work system theory (WST) as a theoretical core and then switches to focusing on CHSs and digital agents, both of which are special cases with some properties not shared by work systems in general. Examples of CHSs and related digital agents include health monitoring CHSs that use

wearable devices, hospital CHSs that use digital agents for suggestions and alarms, factory production CHSs that people control with the help of dashboards, and design CHSs in which engineers use digital agents to specify and evaluate artifacts. Excluded from this discussion are implanted devices such as pacemakers that are not used consciously and imperceptible digital agents that are embedded in devices such as network controllers or dishwashers.

Goal. This paper's central premise is that thinking broadly about CHSs leads to IS engineering ideas that are not at the core of current IS engineering approaches. Those ideas are related to topics such as dimensions of smartness, facets of work, roles and responsibilities of digital agents, patterns of interaction between humans and digital agents, different types of knowledge objects that may be created and/or used, and a broad range of evaluation criteria. This paper shows how those and other ideas can be integrated into IS engineering for CHSs.

Organization. The next section identifies and defines basic concepts including work, work system, work system theory, CHS, digital agent, and IS engineering. A table uses the nine elements of the work system framework to identify CHS-related challenges that might lead to incorporating new ideas into IS engineering. The remainder of the paper focuses on ideas that could help in addressing some of those challenges. Those ideas include dimensions of smartness for systems and devices, facets of work, roles and responsibilities of digital agents, patterns of interaction between people and devices, different types of knowledge objects, and criteria for evaluating CHSs. In combination, those ideas and others point to important issues that IS engineering for CHSs ideally should address in attempts to make CHSs smarter.

2 Basic Definitions

2.1 Work

In relation to purposeful systems, work is defined as the use of resources to produce product/services for human or nonhuman customers or for oneself. Those resources include human, informational, physical, financial, and other types of resources. Work involves activities that try to be productive and that may occur in any business, societal, or home setting. This definition of work is not directly related to careers, jobs, or business organizations.

2.2 Work Systems

This term appeared in the first edition of *MIS Quarterly* [1] and is a natural unit of analysis for thinking about systems in organizations. Work systems (WSs) are described by the three elements of work system theory (WST – [2]). WST (Fig. 1) include the definition of WS, the work system framework, and the work system life cycle model. The first *and/or* in the definition of WS addresses trends toward automation of work by saying that WSs may be sociotechnical systems (with human participants doing some of the work) or totally automated systems. Many of the same WS properties

apply equally to sociotechnical WSs and totally automated WSs regardless of the extent to which machines in those systems are viewed as smart. The work in WSs may be structured to varying degrees, e.g., unstructured (designing a unique advertisement), semi-structured (diagnosing an ambiguous medical condition), workflows (processing invoice payments), or highly structured (manufacturing semiconductors).

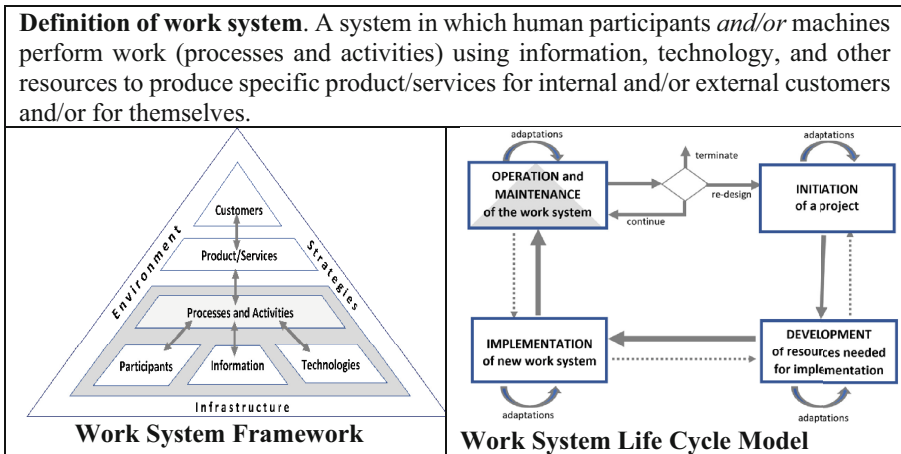


Fig. 1. Three components of work system theory

Work System Framework. This framework in Fig. 1 identifies nine elements of a basic understanding of a WS’s form, function, and environment during a period when it retains its identity even as incremental changes may occur. *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. (Thus, customers are not viewed as part of a WS’s environment.) *Environment, infrastructure, and strategies* are external to the WS even though they have direct impacts on its operation.

Core components of the work system framework have appeared in the BPM literature at least since 2005, when a BPR framework in a paper on best practices in business process redesign [3] cited a “work centered analysis framework” from a 1999 IS textbook and included *customers, products, information, technology, business process, and organization* (consisting of structure and “population”). In 2021, [4] extended that BPR framework by proposing a redesign space for exploring process redesign alternatives. Five of six layers of that redesign space are core elements of the work system framework in Fig. 1, (*customer, product/service, business process, information, technology*); the other layer replaces *participants* with organization.

Information Systems and Other Special Cases. 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 [5] and was one of 34 definitions of IS noted in [6]. 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 changes in the IS affect that WS. Projects, service systems, self-service systems, and some supply chains (interorganizational WSs) are other special cases. E.g., software development projects and projects that create or customize a machine are WSs that produce specific product/services and then go out of existence.

2.3 Human Agency and Workarounds

Human WS participants in sociotechnical WSs may not perform work exactly as designated by designers or managers. The deviations may stem from mistakes, from lack of training, or from intentional workarounds [7–10] in which WS participants deviate from expected practices when they encounter obstacles that prevent them from achieving organizational or personal goals. A related issue is intentional nonconformance with process specifications or expectations [11, 12]. Workarounds and nonconformance generate significant challenges for process mining [13, 14]. Ideally, IS engineering should try to anticipate mistakes, workarounds, and other types of nonconformance that might be possible to predict.

2.4 Digital Agents

The concept of digital agent is useful for understanding cyber-human systems. An agent is an entity that performs task(s) delegated by another entity. An algorithmic agent is a physical or digital agent that operates by executing algorithms, i.e., specifications for achieving specified goals within stated or unstated constraints by applying specific resources such as data inputs. Algorithmic agents are WSs because they perform work using information, technologies, and other resources to produce product/services for their direct customers, which may be people or non-human entities. A digital agent is a totally automated IS that is an algorithmic agent serving a specific purpose. A gigantic ERP system would not be viewed as a digital agent because it is better described as IT infrastructure that serves needs of many different WSs with many different customers.

2.5 Cyber-Human Systems

Cyber-human systems (CHSs) can be viewed as sociotechnical WSs in which human participants and digital agents interact in nontrivial ways that involve complex attention by human participants. For example, using a digital twin to try to solve a difficult problem is a nontrivial interaction, quite different from simply selecting a document from a list. Much of this paper uses the abbreviation CHS to emphasize its focus on CHSs rather than

other sociotechnical WSs that may not involve nontrivial interactions between people and software-driven devices.

Alternative Configurations of Cyber-Human Systems. CHSs occur along a continuum involving different forms of initiation of interactions between people and machines. Alternatives along that spectrum of arrangements can be described as machine-in-the-loop, mixed initiative interactions, and human-in-the-loop.

With machine-in-the-loop (e.g., [15]), CHS participants guide the operation of the CHS and interact with digital agents to request status or performance information or to obtain suggestions concerning decisions. With mixed initiative interactions, either CHS participants or digital agents may take the initiative. For example, a digital agent monitoring a process might issue a warning and might request a response from a CHS participant about whether corrective action is needed. From the other side, a CHS participant might initiate an interaction to request a nonstandard action or status report from the digital agent. With human-in-the-loop interactions (e.g., [16, 17]), the digital agent performs activities autonomously, but requests confirmation or instructions from CHS participants when it encounters situations requiring human judgment.

2.6 IS Engineering

IS engineering is a CHS involving the development, evaluation, and improvement of specifications of systems devoted to processing information, thereby supporting, controlling, or executing activities that stakeholders recognize and care about. Interactions between WS participants and digital agents in IS engineering may lie anywhere along the spectrum of CHS arrangements that goes from machine-in-the-loop to mixed initiative interactions to human-in-the-loop.

3 Cyber-Human System Challenges Related to Elements of the Work System Framework

Table 1 uses the elements of the work system framework (Fig. 1) to identify some of the IS engineering challenges related to making CHSs smarter. Subsequent sections of this paper present ideas that help in addressing some these challenges.

Table 1. IS engineering challenges related to making cyber human systems smarter

Work system element	Challenges related to making cyber-human systems smarter
Technologies	<ul style="list-style-type: none"> • Providing technologies whose use increases the smartness of a CHS regardless of the extent to which the technologies themselves are smart • Integrating smarter and less smart technologies within a CHS
Information	<ul style="list-style-type: none"> • Extending the idea of information in CHSs to include explicit knowledge that smarter cyber-human systems can use
Participants	<ul style="list-style-type: none"> • Anticipating and adjusting for predictable mistakes, workarounds, or other reasons why people may or may not perform work in exactly the way that designers or managers imagined it would be performed • Anticipating and adjusting for human sensitivity issues in CHSs, e.g., people may dislike performing work in which machines play a major role, especially when the machines monitor human work closely or tell people what to do
Processes and activities	<ul style="list-style-type: none"> • Finding effective ways to integrate CHSs into activities in processes that may range from unstructured to semi-structured to highly structured • Finding effective ways to support different facets of work in CHSs, such as making decisions, communicating, coordinating, and many other facets of work that may present other challenges • Making effective use of different possibilities for interaction patterns between WS participants and digital agents within CHSs
Product/services	<ul style="list-style-type: none"> • Making product/services smarter so that they provide greater benefit for customers while fitting within work practices and contexts of customers
Customers	<ul style="list-style-type: none"> • Recognizing that customers often are involved in activities in CHSs and may not be trained about capabilities and quirks of digital agents or other parts of the CHS
Environment	<ul style="list-style-type: none"> • Recognizing that the surrounding competitive, political, legal, organizational, and technological environments may present challenges that seem far removed from the operation of specific CHSs and digital agents
Infrastructure	<ul style="list-style-type: none"> • Recognizing that human, informational, and technical infrastructures may not be adequate for CHSs that use smarter digital agents
Strategies	<ul style="list-style-type: none"> • Recognizing that strategies related to advanced technology uses may be little more than slogans

4 Dimensions of Smartness for Systems and Devices

Supporting the development of smarter CHSs is a possible aspiration for IS engineering. An approach to smartness of devices and systems is explained in [18], which says that the smartness of purposefully designed entity X is a design variable related to the extent

to which X uses physical, informational, technical, and/or intellectual resources for processing information, interpreting information, and/or learning from information that may not be specified by its designers. That definition leads to four broad categories of smartness that are each related to 5 to 7 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.

The smartness built into a device or system for any of the above capabilities can be characterized along the following dimension [18]:

- **Not smart at all.** Does not perform activities that exhibit the capability.
- **Scripted execution.** Performs activities related to a specific capability 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.

The four categories of smartness, the various capabilities within each category, and the related dimensions for describing smartness reveal that most “smart” entities in current use are not very smart after all because they perform only scripted execution for a limited number of capabilities. For example, many current uses of machine learning are built on transforming a dataset into an algorithm for making choices or predictions, but have no ability to process other kinds of data and very limited ability to take action in the world, self-regulate, or acquire new knowledge autonomously. Similarly, many nominally smart devices can capture data and use it in a scripted way, but cannot perform other kinds of activities at a level that seems “smart,” especially when compared with the smartness of sentient robots in science fiction.

5 Facets of Work

One possible approach for trying to make a CHS smarter is to increase the smartness of individual facets of work within that CHS. Facets of work are aspects of work that can be observed or analyzed, such as making decisions, communicating, processing information, and coordinating. The idea of facets of work is an extension of WST that

is useful for describing and analyzing the use of a digital agent by a CHS. That idea grew out of research for bringing richer and more evocative concepts to systems analysis and design to facilitate interactions between analysts and stakeholders, as is explained in [19]. The notion of “facet” is an analogy to how a cut diamond consists of a single thing with many facets that can be observed or analyzed. Psychology, library science, information science, and computer science have used the idea of facets, but with quite different meanings and connotations.

All 18 of the facets of work in Table 2 apply to all CHSs, are associated with specific concepts, bring evaluation criteria and design trade-offs, have sub-facets, and bring open-ended questions for analysis and design [19]. Some facets overlap in many situations (e.g., making decisions and communication). The iterative design process that led to selection of the 18 facets might have led to a different set of facets, perhaps 14 or 27. Determination of whether or not to include a type of activity as one of the 18 facets of work in [19] was based on the extent to which that type of activity was associated with a nontrivial set of concepts, evaluation criteria, design trade-offs, sub-facets, and open-ended questions that could be useful in analysis and design. In relation to digital agents, facets of work provide a way to be specific about requirements for many types of capabilities that might be overlooked otherwise.

Table 2. 18 facets of work [19].

Making decisions	Communicating	Providing information
Representing reality	Learning	Coordinating
Performing physical work	Providing service	Applying knowledge
Planning	Improvising	Performing support work
Creating value	Thinking	Controlling execution
Processing information	Interacting socially	Maintaining security

There is no assumption that the facets of work should be independent. To the contrary, the facet *making decisions* often involves other facets such as *communicating*, *learning*, and *processing information*. The main point is that each facet can be viewed as part of lens for thinking about where and how CHSs might use digital agents.

6 Roles and Responsibilities of Digital Agents

Another possible direction for finding ways to make CHS smarter is to focus on roles and responsibilities of digital agents within CHSs. Roles that digital agents might play in supporting or performing work in a CHS can be identified along a spectrum from the lowest to the highest degree of direct involvement of the digital agent in the execution of the CHS’s activities. Shneiderman’s human-centered AI (HCAI) framework [20, 21] was a starting point for developing a spectrum of such roles. That framework has two dimensions: low vs. high computer automation and low vs. high human control. Deficiency or excess along either dimension may lead to worse results for organizations, for

CHS participants, and/or for customers. For current purposes, however, the low vs. high distinctions in those dimensions provide too little detail to inspire vivid visualization and discussion of how or why a digital agent might be applied in a CHS's operation or might affect its stakeholders.

An iterative attempt to expand on Shneiderman's HCAI framework to make it more useful for detailed description and analysis of WSs focused initially on the low vs. high automation dimension. The three roles were identified initially: support, control, and perform. Trial and error consideration of many familiar examples led to six types of roles that a digital agent might play for a WS (and hence a CHS). Specific instances of each type might support HCAI values and aspirations or might oppose those values and aspirations (e.g., micromanagement or surveillance capitalism). The following comments about the six roles include examples that promote human-centric values and other examples that seem contrary to those values:

Monitor a Work System. A digital agent might monitor and measure aspects of work to assure that a CHS's processes and activities are appropriate for CHS participants. In some cases the digital agent might generate alarms when digital traces of work start going out of accepted bounds regarding health, safety, and cognitive load. On the other hand, the digital agent might monitor work so closely that people would feel micromanaged or disrespected.

Provide Information. A digital agent might provide information that helps people achieve their work goals safely and comfortably without infringing on privacy and other rights of people whose information is used. On the other hand, the digital agent might provide real time comparisons that lead to toxic levels of competition between workers.

Provide Capabilities. A digital agent might provide analytical, visualization, and computational capabilities that help CHS participants achieve their work objectives safely and with appropriate effort. On the other hand, new digital agent capabilities might erode or eliminate the importance of skills that CHS participants had developed over many years (e.g., de-skilling of insurance underwriters by partial automation).

Control Activities. A digital agent might control CHS activities directly to prevent specific activities from going out of bounds related to worker safety, time on the job, stress, and other variables that can be measured and used to control a CHS. On the other hand, a digital agent's frequent feedback about performance gaps (e.g., rate of call completions in support centers) might increase anxiety about whether goals can be met.

Coproduce Activities. A digital agent might be deployed in a division of responsibility in which the digital agent and human CHS participants have complementary responsibilities for performing their parts of the work. In some instances the initiative for the next step might shift back and forth between the digital agent and the CHS participants depending on the status of the work. On the other hand, giving the digital agent a leading role might leave some CHS participants feeling that they are working for a machine.

Execute Activities. A digital agent might execute activities that should not or cannot be delegated to people. For example, a digital agent might perform activities that are difficult, dangerous, or impossible for people to perform as the CHS produces product/services. On the other hand, a digital agent might automate activities that people

could perform more effectively due to their ability to understand and evaluate exceptions and unexpected situations.

6.1 Agent-Responsibility Framework

The agent-responsibility (AR) framework in Fig. 2 [22, 23] combines the facets of work in Table 2 with the six digital agent roles introduced above. It assumes that digital agent usage occurs when a digital agent performs one or more roles (the horizontal dimension) related to one or more facets of work (the vertical dimension). For the sake of easy visualization, the abbreviated version of the AR framework in Fig. 2 uses only 6 of the 18 facets in Table 2.

Combining the AR framework’s two dimensions leads to pinpointing design issues concerning the extent to which a digital agent should have responsibilities involving roles related to facets of work, e.g., monitoring decisions, providing capabilities used in coordinating, or performing security-related activities automatically. The AR framework can be used in many ways. For example, [22] and [23] show how different roles might apply to many different facets of work in CHSs such as a hiring system, use of an electronic medical records system, an ecommerce system, and in other work systems such as an automated auction and the IS that partly controls a self-driving car.

Facet of work	^^	Making decisions						
	>>	Communicating						
		Processing information						
		Coordinating						
	<<<	Creating value						
	<<<	Maintaining security						
		Monitoring the CHS	Providing information	Providing capabilities	Controlling activities	Coproducing activities	Executing activities	
		<<<<<<<< Spectrum of roles >>>>>>>>						

Fig. 2. (Abbreviated) Agent responsibility framework with six roles and six facets of work

7 Patterns of Interaction Between CHS Participants and Digital Agents

Yet another approach for increasing the smartness of an CHS is to improve interactions between CHS participants and digital agents. Those interactions may include unidirectional, mutual, or reciprocal actions, effects, relationships, and influences. IS engineering can look at those interactions by adapting system interaction patterns from preliminary

research [24] that used four categories to organize 19 patterns of interaction between work systems. The idea of system interaction pattern was inspired by an analogy to software design patterns (solutions to recurring software problems), an idea that was inspired by design patterns for architecture [25]. Features of software design patterns include name, intent, problem, solution, entities involved in the pattern, consequences, and implementation, i.e., a concrete manifestation of a pattern [26].

Table 3 uses four categories to organize 16 or the 19 *system interaction patterns* that [24] explains. (The other three are not relevant for interactions between CHS participants and digital agents.) In specific situations, instances of most of the 16 interaction patterns can be described in detail in terms of roles and responsibilities of the CHS participant and digital agent, cause or trigger of the interaction, desired outcome, typical process during the interaction, and alternative versions of that process. Occasionally relevant aspects of those interactions include constraints, risks and risk factors, byproducts, and verification of interaction (important in some cases, but not all). The interaction patterns in Table 3 do not need the rigor or specificity of software design patterns because they serve more as a map for identifying different types of interaction situations and related issues that need to be addressed.

Table 3. Sixteen Patterns of Interaction between CHS Participants and Digital Agents

<p>One-way patterns (unidirectional)</p> <ul style="list-style-type: none"> • Inform • Command • Request • Commit • Refuse 	<p>Co-production patterns (bilateral)</p> <ul style="list-style-type: none"> • Converse • Negotiate • Mediate • Share resource • Supply resource
<p>Access and visibility patterns</p> <ul style="list-style-type: none"> • Monitor • Hide • Protect 	<p>Unintentional impact patterns</p> <ul style="list-style-type: none"> • Spillover • Indirect interaction • Accidental interaction

One-way patterns are unidirectional interactions that have been studied in relation to the language action perspective (LAP). These patterns include *inform*, *command*, *request*, *commit*, and *refuse*, all of which appeared in a study of email [27]. Examples show how all five might occur: the digital agent might *inform* the CHS participant about a condition that requires action; the digital agent might *command* the CHS participant to take a particular action that is necessary to avoid a breakdown or for other reasons; the digital agent might *request* that the CHS participant to take a particular action that seems appropriate; the digital agent might *commit* to taking a particular action that depends on a nonobvious aspect of its internal state; the digital agent might *refuse* to take a particular action, perhaps because that action is infeasible due to its internal state.

Coproduction patterns are bilateral patterns whose jointly produced interactions involve multiple 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. The

other two are fundamentally about coordination as described by coordination theory [28, 29]. For example, the digital agent and CHS participant might *converse* or *negotiate* about how to proceed toward achieving a goal; the digital agent might *mediate* in activities involving several CHS participants; the digital agent and CHS participant might *share* a physical resource; the digital agent might *supply* the CHS participant with a resource by providing convenient access to a tool.

Access and visibility patterns are unidirectional interaction patterns through which one entity achieves or blocks access to resources that may include information about the state of either entity. These patterns include *monitor*, *hide*, and *protect*. For example, the CHS participant might *monitor* the state of the digital agent to make sure that it is fully operable. The CHS participant might *hide* information that might otherwise cause the digital agent to malfunction, e.g., a workaround that might be necessary when an exception condition dictates that typically expected interactions are not appropriate. A CHS participant might *protect* a digital agent by intentionally modifying data that that otherwise might cause it to malfunction due to limits in its algorithms.

Unintentional impact patterns may occur in *spillover*, *indirect*, and *accidental* interactions. These are the least articulated patterns because of the great uncertainty about the sources and effects of many unintentional impacts. A *spillover* interaction could occur when a CHS participant's inability to complete a task requires the digital agent to operate in a nonstandard or unplanned manner. An *indirect* interaction occurs when a mistake by a CHS participant creates a situation that requires corrective action by the digital agent. An *accidental* interaction occurs when a CHS participant's behavior accidentally causes the digital agent to stop operating. Unintentional impacts often are difficult to anticipate, but ignoring the possibility that they will occur certainly is not beneficial for either CHS design or CHS operation.

An attempt to make a CHS smarter might involve changing elements of typical interaction patterns in any of the first three categories. Those elements include actor roles (e.g., requestor/respondent, initiator/recipient, partner, or intermediary), actor type (e.g., CHS participant or digital agent), actor rights for each role, actor responsibilities for each role, cause or trigger of the interaction, desired outcome, expected process for the interaction, possible states of an interaction, and alternative enactments. Other elements of interaction patterns that sometimes might point to paths for improvement include constraints, risks and risk factors, relevant concepts, verification of completion, and evaluation of the interaction.

8 Knowledge Objects

Information processing and knowledge acquisition are two of the categories of smartness identified earlier. The taxonomy of knowledge objects (KOs) in Fig. 3 points to ways to make CHSs and digital agents smarter even though it is an extension of ideas developed for a different purpose. That purpose was describing science as the creation, evaluation, accumulation, dissemination, synthesis, and prioritization of KOs, including the reevaluation, improvement, or replacement of existing KOs by other KOs that are more effective for understanding aspects of the relevant domain.

In relation to current concerns, human CHS participants and digital agents have quite different capabilities in regard to the KOs in the taxonomy in Fig. 3. People can make

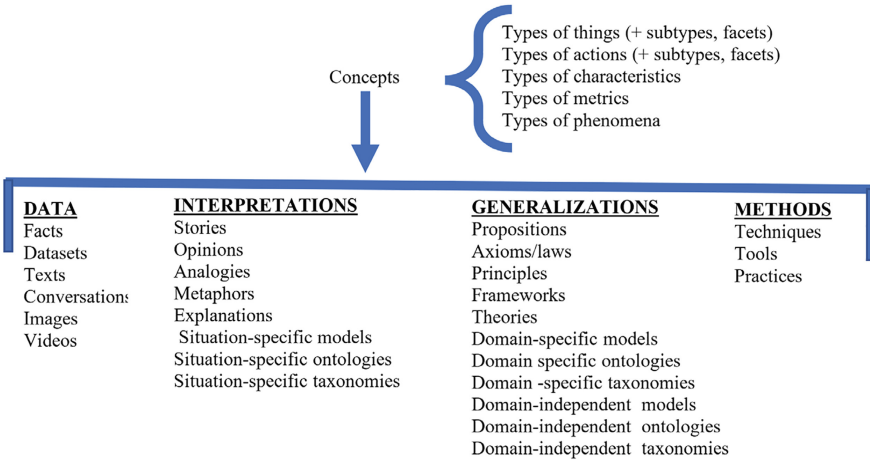


Fig. 3. Taxonomy of knowledge objects (revision of a figure in [30])

explicit use of abstract KOs that Fig. 3 categorizes as types of generalizations or methods. For example, when confronted with issues or obstacles they can think about which concepts, principles, or theories might be helpful. In contrast, most current digital agents and computerized components of other parts of CHSs can only process data (the non-abstract category in Fig. 3). Without engaging in debates about the strengths and limits of current machine learning algorithms and generative AI capabilities (e.g., producing texts and images starting from user prompts), it seems fair to say that greater ability to process and apply abstractions that express generalizations or methods might make CHSs smarter. [31] suggests that activities in the human parts of CHSs might become smarter if the human CHS participants had greater access to codified knowledge, possibly through knowledge graphs. In contrast, it might be possible to make digital agents smarter by embedding capabilities for processing and applying abstractions. Both aspirations might lead to beneficial results, but the path toward those capabilities is quite unclear.

9 Evaluating Cyber-Human Systems and Digital Agents

Both CHSs and digital agents should be evaluated based on multiple criteria that address different types of issues. Big picture criteria such as efficiency, effectiveness, equity, engagement, empathy, explainability, exceptions, and externalities (the 8 E’s) [32] apply to most nontrivial CHSs and to many digital agents that they use. The first two evaluation criteria are fundamentally about how well a CHS or digital agent achieves its operational goals. The other six are not as directly linked to operational goals but often affect CHS performance and/or perceptions of product/services that a CHS produces. Beyond the 8 E’s, a full picture of the performance of a CHS or digital agent requires metrics related to individual elements of the work system framework (e.g., accuracy and timeliness of

the information in the CHS). Different metrics for the various work system elements have been discussed elsewhere, but only the 8 E's will be discussed here.

Efficiency. Both a CHS and a digital agent that supports it should be efficient and should support the efficiency of related work systems, i.e., the minimally wasteful production of the product/services of those work systems.

Effectiveness. Both a CHS and a digital agent that supports it should meet or exceed expectations regarding effectiveness in satisfying needs and expectations of their customers, which are often other work systems.

Equity. Both a CHS and a digital agent that supports it should operate in ways that are fair to stakeholders including CHS participants, CHS customers, and others affected directly and indirectly. Equity often presents challenges because designers, managers, and others may be unaware of their own biases and biases built into the CHS and/or the digital agent.

Engagement. CHSs and digital agents should engage CHS participants wherever that might maximize benefits from their insights or might make their work environments healthier, more satisfying, and more productive.

Empathy. CHSs and digital agents should reflect realistic consideration of the goals, capabilities, health, and comfort of CHS participants and customers that use the CHS's product/services. Lack of such empathy could have negative impacts on CHS participants, on the CHS's operational performance, on product/services that it produces, or on customers who receive and use its product/services.

Explainability. Both a CHS and a digital agent that supports it should be understandable by people who are affected by it and/or by product/services that it produces. This issue has been discussed widely in regard to AI applications whose outputs cannot be linked in an understandable way to inputs related to individuals, groups, or situations. Inadequate explainability results in confusions, errors, misuse of product/services, and possible harm to people who are affected directly or indirectly.

Exceptions. Frequent exceptions challenge many real world processes. Those challenges are amplified in CHSs because exceptions may come from the environment or from mistakes by participants or digital agents in the CHS.

Externalities. Current attention to sustainability is a strong reminder that the evaluation of both CHSs and digital agents should also consider identifiable externalities that may affect people or property not directly involved with the WS.

10 Conclusion: Toward Smarter Cyber-Human Systems

This paper defined cyber-human system, identified a series of challenges related to making CHSs smarter (Table 1), and explained how smartness of systems and devices can be described in relation to capabilities for information processing, action in the world, internal regulation, and knowledge acquisition. This paper presented ideas related to some of the challenges in Table 1 but could only cite references related to others, such

as anticipating workarounds, noncompliance, and other impacts of human agency. The ideas covered here suggest the following directions that IS engineering might pursue when attempting to increase the smartness of CHSs:

Use More Powerful Digital Agents. A central purpose of many IS engineering efforts is to create or extend digital agents whose information processing activities enable the CHS to be smarter in terms of action in the world, internal regulation, and/or knowledge acquisition. This path toward smarter CHSs does not require impressively smart digital agents. It only requires scripted execution of programs that capture, transmit, store, retrieve, delete, manipulate, and/or display information that increases CHS capabilities related to any of the multiple dimensions of smartness.

Extend Roles and Responsibilities of Digital Agents. The discussion of facets of work, roles of digital agents, and responsibilities of digital agents in relation to facets of work identifies possible directions for making a CHS smarter by increasing its capabilities related to any of the digital agent roles (the horizontal axis of the AR framework) and any of the 18 facets of work in Table 3 (the vertical axis of the AR framework). IS engineering also could pursue those possibilities through more focused attention to the AR framework, e.g., how could digital agents with specific roles improve results for specific facets of work. It is easy to imagine interactive tools that might display the roles, facets of work, or combinations that would help in identifying roles and responsibilities that could be executed more successfully, and possibly in a smarter manner.

Improve Interactions Between Digital Agents and Other Parts of CHSs. The big picture choices of machine-in-the-loop, mixed initiative interaction, and human-in-the-loop plus the 16 interaction patterns in Table 3 point to many possibilities that IS engineering might pursue for making CHSs smarter. In all cases, the question is not whether machine-in-the-loop, mixed initiative, or human-in-the-loop is better in general. The key question is whether changes in the form and details of interactions between CHS participants and digital agents will result in better CHS capabilities, especially greater flexibility that might lead to greater smartness.

Introduce and Use New Knowledge Objects. The different types of knowledge objects in Fig. 3 raise the possibility that IS engineering might make CHSs and/or digital agents smarter by enabling their direct use of abstract knowledge objects such as principles, theories, models, and methods. Efforts directed toward explicit use of abstract knowledge objects would try to open new possibilities because current IS engineering methods focus primarily on pre-defined processing of pre-defined data.

Use the Eight Evaluation Criteria. IS engineering uses the criteria of efficiency and effectiveness routinely in evaluating CHSs and digital agents. Greater attention to the other six criteria might provide an impetus toward smarter CHSs and/or digital agents. Greater attention to equity, engagement, empathy, and explainability could lead to enhancing CHS smartness through fuller use of human rather than machine intelligence. Greater attention to exceptions and externalities might help in seeing the limits of efficiency and effectiveness as overriding criteria for action in the world.

References

1. Bostrom, R.P., Heinen, J.S.: MIS problems and failures: a socio-technical perspective, part II: the application of socio-technical theory. *MIS Q.* **1**(1), 11–28 (1977)
2. Alter, S.: Work system theory: overview of core concepts, extensions, and challenges for the future. *J. Assoc. Inf. Syst.* **14**(2), 72–121 (2013)
3. Reijers, H.A., Mansar, S.L.: Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics. *Omega* **33**(4), 283–306 (2005)
4. Gross, S., et al.: The business process design space for exploring process redesign alternatives. *Bus. Process. Manag. J.* **27**(8), 25–56 (2021)
5. Alter, S.: Defining information systems as work systems: implications for the IS field. *Eur. J. Inf. Syst.* **17**(5), 448–469 (2008)
6. Boell, S.K., Cecez-Kecmanovic, D.: What is an information system? In: Proceedings of HICSS (2015)
7. Alter, S.: Theory of workarounds. *Commun. Assoc. Inf. Syst.* **34**(55), 1041–1066 (2014)
8. Röder, N., Wiesche, M., Schermann, M., Krömer, H.: Why managers tolerate workarounds—the role of information systems. In: Proceedings of AMCIS (2104)
9. Alter, S.: A workaround design system for anticipating, designing, and/or preventing workarounds. In: Gaaloul, K., Schmidt, R., Nurcan, S., Guerreiro, S., Ma, Q. (eds.) CAISE 2015. LNBIP, vol. 214, pp. 489–498. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19237-6_31
10. Beerepoot, I., Van De Weerd, I.: Prevent, redesign, adopt or ignore: improving healthcare using knowledge of workarounds. In: Proceedings of ECIS (2018)
11. Alter, S.: Beneficial noncompliance and detrimental compliance: expected paths to unintended consequences. In: Proceedings of AMCIS (2015)
12. Mertens, W., Recker, J., Kohlborn, T., Kummer, T.: A framework for the study of positive deviation in organizations. *Deviant Behav.* **37**(11), 1288–1307 (2016)
13. Outmazgin, N., Soffer, P.: Business process workarounds: what can and cannot be detected by process mining. In: Nurcan, S., et al. (eds.) BPMS/EMMSAD -2013. LNBIP, vol. 147, pp. 48–62. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38484-4_5
14. Wijnhoven, F., Hoffmann, P., Bemthuis, R., Bokseveld, J.: Using process mining for workarounds analysis in context: Learning from a small and medium-sized company case. *Int. J. Inf. Manag. Data Insights* **3**(1), 1–15 (2023)
15. Clark, E., et al.: Creative writing with a machine in the loop: case studies on slogans and stories. In: Proceedings of 23rd International Conference on Intelligent User Interfaces, pp. 329–340 (2018)
16. Munir, S., et al.: Cyber physical system challenges for human-in-the-loop control. In: 8th International Workshop on Feedback Computing (2013)
17. Smith, A., et al.: Closing the loop: user-centered design and evaluation of a human-in-the-loop topic modeling system. In: 23rd International Conference on Intelligent User Interfaces, pp. 293–304 (2018)
18. Alter, S.: Making sense of smartness in the context of smart devices and smart systems. *Inf. Syst. Front.* **22**(2), 381–393 (2019). <https://doi.org/10.1007/s10796-019-09919-9>
19. Alter, S.: Facets of work: enriching the description, analysis, design, and evaluation of systems in organizations. *Commun. Assoc. Inf. Syst.* **49**(13), 321–354 (2021)
20. Shneiderman, B.: Human-centered artificial intelligence: reliable, safe & trustworthy. *Int. J. Hum.-Comput. Interact.* **36**(6), 495–504 (2020)
21. Shneiderman, B.: Human-centered artificial intelligence: three fresh ideas. *AIS Trans. Hum.-Comput. Interact.* **12**(3), 109–124 (2020)

22. Alter, S.: Agent Responsibility Framework for Digital Agents: Roles and Responsibilities Related to Facets of Work. In: Augusto, A., Gill, A., Bork, D., Nurcan, S., Reinhartz-Berger, I., Schmidt, R. (eds.) *Enterprise, Business-Process and Information Systems Modeling*, pp. 237–252. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-07475-2_16
23. Alter, S.: Responsibility Modeling for Operational Contributions of Algorithmic Agents. *AMCIS* (2022)
24. Alter, S.: System interaction patterns. In: *Proceedings of the Conference on Business Informatics* (2016)
25. Alexander, C.: *A Pattern Language: Towns, Buildings, Construction*. Oxford University Press, Oxford (1977)
26. Shalloway, A., Trott, J.R.: *Design Patterns Explained: A New Perspective on Object-Oriented Design*. Pearson Education (2004)
27. Cohen, W.V., Carvalho, V.R., Mitchell, T.M.: Learning to classify email into “speech acts”. In: *EMNLP*, pp. 309–316 (2004)
28. Malone, T.W., Crowston, K.: The interdisciplinary study of coordination. *ACM Comput. Surv.* **26**(1), 87–119 (1994)
29. Crowston, K., Rubleske, J., Howison, J.: Coordination theory: a ten-year retrospective. In: Zhang, P., Galletta, D. (eds.) *Human-Computer Interaction in Management Information Systems*, Vol. I. M. E. Sharpe (2006)
30. Alter, S.: Taking different types of knowledge objects seriously: a step toward generating greater value from is research. *ACM SIGMIS Database Adv. Inf. Syst.* **51**(4), 123–138 (2020)
31. Alter, S.: Extending a Work System Metamodel Using a Knowledge Graph to Support IS Visualization and Development. *ER Forum* (2022)
32. Alter, S.: How can you verify that i am using AI? Complementary frameworks for describing and evaluating AI-based digital agents in their usage contexts. In: *HICSS* (2023)