A Multi-type Artifact Framework for Cyber-Physical, Social Systems Design and Development

Zakaria Maamar[1], Ejub Kajan[2], Mohammed Al-Khafajiy[3], Murtada Dohan[4], Amjad Fayoumi[5], and Fadwa Yahya[6]
[1]University of Doha for Science and Technology, Doha, Qatar, zakaria.maamar@udst.edu.qa
[2] State University of Novi Pazar, Novi Pazar, Serbia
[3] University of Lincoln, Lincoln, U.K.
[4] Northampton University, Northampton, U.K.
[5] Lancaster University, Lancaster, U.K.
[6] Prince Sattam Bin Abdulaziz University, Al kharj, KSA

Abstract

This paper discusses the design and development of cyber-physical, social systems using a set of guidelines that capture the conceptual and technical characteristics of such systems. These guidelines are packaged into a framework that resorts to the concept of artifact. Because of these characteristics, the framework's artifacts are specialized into 3 types referred to as data, thing, and social, all connected together through a set of situational relations referred to as work-with-me, work-for-me, back-me, and avoid-me. To mitigate conflicts blue that could arise because of artifacts' respective time availabilities when they jointly participate in situational relations, policies are put in place defining who does what, when, where, and why. To demonstrate the technical doability of the multi-type artifact framework, a system capturing cyber, physical, and social interactions in a healthcare case-study is developed, deployed, and evaluated.

Keywords: Artifact, Cyber-physical, social system, Policy, Relation.

1. Introduction

It is safe to assume that many transformation "waves" have shaped the Information and Communication Technology (ICT) landscape since the beginning of the 21^{st} century. Worth mentioning are Web 2.0 (*aka* social media) and Internet-of-Things (IoT) waves that are questioning every single software engineering practice related for instance, to requirement elicitation, program testing, and system deployment.

On the one hand, Web 2.0 has motivated companies to abandon top-down commands and bottom-up feedback in order to foster social relations/interactions that would cross all authority levels and occur in all directions. This communication shift reduces cost, improves efficiency, facilitates innovation, among other benefits Turban et al. (2011). Web 2.0 exemplified with social networks, Wikis, and blogs, has allowed companies to reach out to more customers, open new communication channels with stakeholders, and embrace the latest ICT advances and gadgets Badr and Maamar (October 2009). The world has become "social".

On the other hand, IoT has made anything and everything (e.g., white goods, embedded systems, and wrist devices) reachable and connectable ensuring a personalized control of users' cyber-physical surroundings. According to Statista[1], 9.7 billion connected things were in use in 2020 and will reach 29.4 billion by 2030. It is also predicted that the total economic impact of IoT will be between \$3.9 trillion and \$11.1 trillion per year by 2025 DZone (2017 (visited in May 2017). Today's things are ubiquitous producing massive amounts of data about other "things" like vegetable freshness in a storage facility, number of vehicles on a highway, patients' vitals in medical wards, etc.

Although the literature refers to a good number of design and development approaches that are typically meant for systems residing in a cyberspace Jakubowska (2011); Yilma et al. (2021); Zeng et al. (2020), Web 2.0 and IoT are enacting a new generation of systems that this time reside in what we would refer to as cyber-physical, social space. In this "revamped" space full of new opportunities and challenges, the social dimension exposes the limited control of end-users, spontaneity of end-users, anonymity of end-users, and diversity and richness of social content Burégio et al. (2016), whereas the physical dimension exposes IoT devices' reduced size, restricted connectivity, extended mobility, limited energy, and constrained storage. To tap into these opportunities and tackle these challenges, we suggest a framework for the design and development of systems residing in a cyber-physical, social space. The framework resorts to the concept of artifact that we specialize into Data Artifacts ($\mathcal{DA}$) Maamar et al. (2010), Social Artifacts ($\mathcal{SA}$) Maamar et al. (2017), and Thing Artifacts ($\mathcal{TA}$). While $\mathcal{DA}$ are commonly used for modeling data-centric business processes Maamar et al. (2020); Friedow et al. (2018); Nigam and Caswell (2003), the use of $\mathcal{SA}$ and $\mathcal{TA}$ remains limited to a few initiatives like those reported in Maamar (2022) and Maamar et al. (2017). We present how all these artifacts put together would guide the design and development of systems residing in a cyber-physical, social space. Our contributions are, but not limited to, (*i*) combination of $\mathcal{DA}$, $\mathcal{TA}$, and $\mathcal{SA}$ to design and develop systems residing in a cyber-physical, social space, (*ii*) comparison of $\mathcal{DA}$, $\mathcal{TA}$, and $\mathcal{SA}$ from a functional and non-functional perspectives, (*iii*) definition of collaboration mechanisms between $\mathcal{DA}$, $\mathcal{TA}$, and $\mathcal{SA}$, and (*iv*) technical demonstration of a system capturing cyber, physical, and social interactions in a healthcare case-study. The rest of this paper is organized as follows. Section 2 is an overview of cyber-physical systems, discusses some related works, and defines and contrasts the 3 types of artifacts. Section 3 details the framework's modeling and development steps. The technical validation of the framework is given in Section 4. Finally, future work and conclusion are presented in Sections 5 and 6, respectively.

2. Background

This section briefly defines cyber-physical systems, presents some related works, and finally discusses artifact types as per the existing literature.

2.1. Overview of cyber-physical systems

Cyber-Physical System (CPS) term was initially coined by the US National Science Foundation (NSF) in mid 2000 to describe physically-aware engineered systems that incorporate cyber-components along with weaving these cyber-components' computational and communication capabilities into the physical space Wayne (2007). Because of ICT progress, a new generation of socio-technical systems was born by incorporating social media into CPS Baxter and Sommerville (2011). These systems capture details about the cyber, physical, and social space, and

[1] `www.statista.com/statistics/1183457/iot-connected-devices-worldwide.`

could provide human-centric computation services as well Zeng et al. (2020). Since then, CPS has become an integral part of people's daily activities. Examples include smart-cities, -homes, and -health. Furthermore, new business models emerged like smart manufacturing (*aka* Industry 4.0 and beyond), smart transportation, smart logistics, etc. These systems are capable of interacting with each other in a time- and context-aware manner.

In the literature, smart city is a typical CPS that is commonly used for illustration purposes. Chourabi et al. describe smart city as a large organic system connecting many subsystems and components together Chourabi et al. (2012). A smart city's business model is like an organic system consisting of nerves, brain, sensory organs, and knowledge Mitchell (2007). For instance, cameras and air sensors act as sensory organs that alert a control center via nerves (i.e., wrappers that catch these information) of any car-accident situation in a city tunnel and afterwards, report pollution levels (e.g., CO_2). These information are further processed using accumulated knowledge and brain (e.g., algorithms), so that the car-accident situation is handled.

Applying ICT to manufacturing is a long-term concept known as Computer-Integrated Manufacturing (CIM) with centralized control, limited by factory walls, and with no flexibility, context-awareness, and self-configuration. The new emerging CPS-based concept *"provide collaborative manufacturing by integrating things, computers and humans, ubiquitous, artificial and collective intelligence, as well as explicit and tacit knowledge as a whole"* Yao et al. (2019). Whilst pure machine-based systems tend to automate production by replacing people on the production line, Emmanouilidis et al. emphasize the importance of having people manage this line with significant added-value as a result of the interaction between human and non-human actors Emmanouilidis et al. (2019). The authors argue that new ways of creating decision flows should be identified to achieve foreseen benefits of humans in the production line. This would lead to the so called pancake model of decision making where problems are addressed when and where they arise Schneckenberg (2009).

2.2. Related work

In an early work on CPS design Lee (2008), Lee emphasizes that scaling up simple programs to complex systems like CPS calls for new design requirements. The author argues that *"... even simplest C program is not predictable and reliable in the context of CPS because the program does not express aspects of the behavior that are essentials to the system"*. After discussing current design techniques and technological advances, Lee concludes that core abstractions in computing should completely be revisited to achieve full potential of CPS.

In Hehenberger et al. (2016), Hehenberger et al. discuss CPS's design challenges from 3 perspectives namely, physical processes, computation, and integration. From a physical process perspective, a CPS detects changes in the physical space, but also makes changes to it, whilst the computational perspective refers to embedded systems that are larger and more complex in CPS. From an integration perspective, the authors argue that mixing the physical space and computation together takes place within different disciplines thus, incompatibilities and interruptions during design time are expected to occur.

In Törngren and Grogan (2018), Törngren and Grogan discuss design challenges of CPS focusing because of their complexity. They argue that new techniques should be considered for describing highly varying environments and better understanding of uncertainty and composability. While the former relates to all aspects of life-cycle stages (e.g., actual physical capabilities and changes in environment), the latter requires *"both the integration of cyber and physical components and the integration of technical systems with human counterparts"*.

In Zeng et al. (2020), Zeng et al. discuss some common practices like layered, component-driven, model-driven, and virtual integration approaches for CPS design. The authors suggest a unified representation model for CPS's 3 actors namely, people, cyber space, and physical space. This model is elaborated by humans where physical and data flows are represented as acyclic directed graphs capturing message communications between these 3 actors.

2 recent references provide a stat-of the-art discussion of CPS Systems (CPSS) in terms of definitions, underlining principles, and application areas Yilma et al. (2021) as well as knowledge gaps identification in IoT in healthcare Rejeb et al. (2023). In Yilma et al. (2021), Yilma et al. emphasize that "*a CPSS comprises at least one physical component responsible for sensing and actuation, one cyber component for computations and one social component for actuating social functions*" and argue that current state of research is way behind from achieving the required level of maturity. The authors also argue for CPSS "socialization", a process that will allow IoT devices to serve as social actuators. In Rejeb et al. (2023), Rejeb et al use statistical analysis based on keywords frequency, citations, keyword co-occurrence, and co-citation to identify research gaps in IoT-based healthcare domain. Despite that the given analysis focuses only on IoT, some gaps are common for CPSS, too. They advocate for a single smart system that would consist of communication technologies, sensors and devices, networked applications, and people. In summary, CPSS development is still lagging behind compared to other ICT fields. In this paper we suggest the socialization of CPSS design and development using artifacts and social relations between these artifacts.

2.3. Types of artifacts

Initial thoughts about artifacts refer back to 2 theories, Computers Are Social Actors (CASA, Nass and Moon (2000)) and Actor Network Theory (ANT, Law and Hassard (1999)). Our $\mathcal{DA}$ are actors within a network where $\mathcal{SA}$ and $\mathcal{TA}$ exchange these $\mathcal{DA}$ to achieve some sort of agreed outcome. The following provides further details about each artifact type.

Data artifact. According to Nigam and Caswell Nigam and Caswell (2003), a $\mathcal{DA}$ is a concrete, identifiable, self-describing chunk of information that can be used by a business person to actually run a business, and has a set of states that form the $\mathcal{DA}$'s lifecycle. In Kumaran et al. (2008), Kumaran et al. assist IT practitioners with identifying the necessary $\mathcal{DA}$ for their applications through guidelines. In Narendra et al. (2009), Narendra et al. model business processes using context-based artifacts and Web services. The authors abstract processes using models that are expressive for non-IT practitioners and could be based on $\mathcal{DA}$. In Popova et al. (2015), Popova et al. acknowledge the role of $\mathcal{DA}$ in modeling business processes and propose ways to discover lifecycles of $\mathcal{DA}$. These ways are implemented as software plug-ins for ProM[2], a generic open-source framework for supporting different process mining techniques. In Fig. 1, we illustrate the lifecycles of 3 $\mathcal{DA}$s, order, customer, and bill, that are used to model a simplified version of the traditional purchase-order scenario Narendra et al. (2009). In this figure, rounded rectangles, plain lines, and dashed lines represent artifact states, transitions within the same lifecycle, and messages between states in separate lifecycles, respectively.

Social artifact. In Maamar et al. (2017), the authors define a $\mathcal{SA}$ as a meaningful piece of information that a Web 2.0 application makes available for users and other applications

[2] `www.promtools.org.`

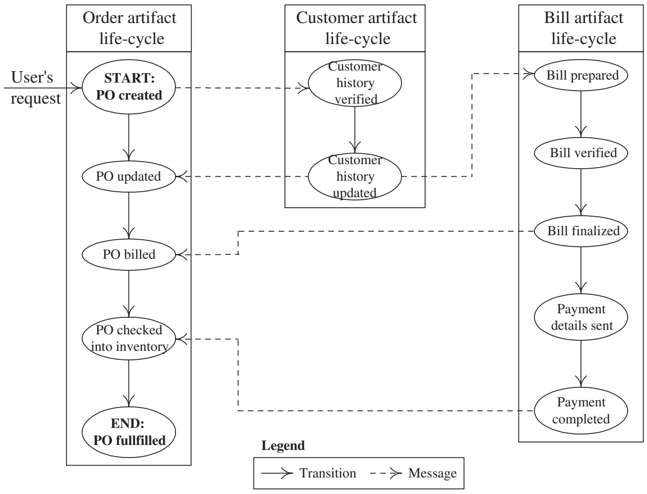

Figure 1: Representation of a $\mathcal{DA}$-based purchase-order BP (adapted from Narendra et al. (2009))

as well (whether Web 2.0 or not). From an operation perspective, a $\mathcal{SA}$ is automatically associated with a social action like post in Facebook and tweet in Twitter. And, from a data perspective, a $\mathcal{SA}$ is associated with the outcome of executing a social action like posting a note on Facebook; this note becomes a $\mathcal{SA}$. By analogy with a $\mathcal{DA}$, a $\mathcal{SA}$ includes a set of descriptive (structured and/or unstructured) data properties, a set of states that identify changes in the $\mathcal{SA}$, and a lifecycle built upon these states. An example of using $\mathcal{SA}$ in social-media mining is presented in Kajan et al. (2020) where a $\mathcal{SA}$ would fall into one of the categories listed in Table 1 namely, *communication*, *sharing*, and *enrichment*.

Table 1: Representative categories of $\mathcal{SA}$ based on social actions (Kajan et al. (2020))

Category	Description	Examples of social actions
Communication	Includes actions that establish back-and-forth interactions between users, which should engage them in joint operations	**Chat** with a user or group of users, **Poke** someone, Send direct messages to a user's inbox
Sharing	Includes actions that establish one-way interactions and allow to create and edit shared content and to facilitate this content's consumption	**Co-author** a text/media on a Wiki, **Publish** a post on a Blog Web site, **Upload** a photo/video on a public repository, or any other data (e.g., sensor reading)
Enrichment	Includes actions that provide additional [meta] data on shared content by providing opinions and/or ranking	**Comment** a post, **Tag** users' photos, videos, activities, etc.

Thing artifact. In Maamar (2022), the authors define a $\mathcal{TA}$ as a chunk of information that refers to its functionality, captures its lifecycle, and tracks its interactions. From an operation perspective, a $\mathcal{TA}$ would automatically be associated with a functionality like sensing CO_2 in a given surrounding. And, from a data perspective, a $\mathcal{TA}$ would be associated with the outcome of executing a functionality like sending the measured value of CO_2 to a dedicated CPS node; this message would become a $\mathcal{DA}$. By analogy with $\mathcal{SA}$, a $\mathcal{TA}$ would include a set of descriptive data properties, a set of states that identify changes in the $\mathcal{TA}$ (e.g., CO_2 detected and CO_2 not detected), and a lifecycle built upon these states.

Table 2: Primitive functionalities of things along with some illustrations

Functionality	Description	Examples of thing functionalities
Sensing	Detects physical phenomena in a surrounding and convert these into electrical impulses	Measuring indoor temperature and share it with devices responsible for temperature regulation
		Detecting air pollutant like CO_2 and $PM10$ and sending measured values to respective thing nodes, e.g., fans responsible for cleaning the air
Actuating	Regulates situations in particular surroundings	Decreasing indoor temperature by adjusting the cooling threshold
		Cleaning ambient air by turning fans on in a given surrounding
		Changing traffic lights due to cameras detecting traffic congestion
Communicating	Sharing messages to relevant parties	Dispatching details on perishable goods

Contrasting artifacts. Despite their different roles in the design and development of cyber-physical, social systems, $\mathcal{DA}$, $\mathcal{SA}$, and $\mathcal{TA}$ present some similarities that we group into 3 main categories (Table 3). These categories are data emphasizing the format, context, privacy, sensitivity, and readiness of data linked to artifacts, stakeholder emphasizing the anonymity and willingness of those linked to artifacts, and others emphasizing the interoperability, resilience, restriction, and security of artifacts.

Table 3: Comparison of artifacts' features

Category	Features	$\mathcal{DA}$	$\mathcal{SA}$	$\mathcal{TA}$
Data	Format: is it known in advance?	y	n	y
	Context: is it known in advance?	y	n	n
	Privacy: is it confidential?	y	n	y
	Sensitivity: is it dependent on any context like time and location?	n	n	y
	Readiness: does it require processing?	n	y	y
Stakeholder	Anonymity: is revealing identity compulsory?	y	n	n
	Willingness: are there incentives?	n	y	y
Others	Interoperability: is collaboration expected?	y	y	y
	Resilience: is failure a serious concern?	y	n	y
	Restriction: is it subject to resource availability?	n	n	y
	Security: is it highly vulnerable?	n	n	y

3. The multi-type artifact framework

After an overview of the framework's key components, the rest of the section details the modeling and development steps that underpin the framework in terms of relations that connect artifacts together and policies that regulate the participation of artifacts in these relations.

3.1. Overview

Fig. 2 illustrates the components of the framework for modeling and developing cyber-physical, social systems. In addition to artifact types, other components include the social space hosting for instance, Web 2.0 applications, and the cyber-physical space hosting for instance, physical sensors and their digital counterparts/representatives. Multiple interactions between the artifacts take place according to these systems' future objectives and requirements. During the monitoring of the social space, $\mathcal{SA}$ produce data that become $\mathcal{DA}$. By analogy with $\mathcal{SA}$, $\mathcal{TA}$ also produce data that become $\mathcal{DA}$ as an outcome of monitoring the cyber-physical space. Finally, $\mathcal{DA}$ are fed back into both the social space and the cyber-physical space allowing the respective participants of these spaces to act accordingly. For instance, a Web 2.0 application relays a social post further and a sensor sees its sensing frequency adjusted.

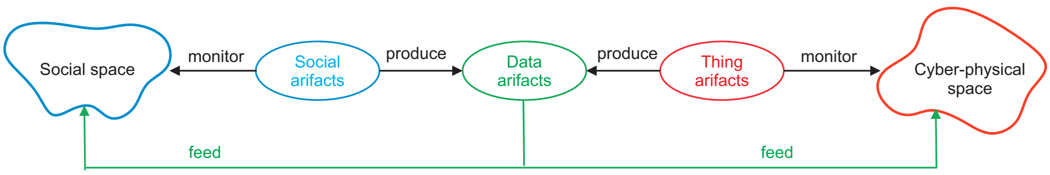

Figure 2: General representation of the multi-type artifact framework

In preparation for detailing the multi-type artifact framework, we back

- the logical existence of $\mathcal{DA}$ with data, $\mathcal{SA}$ with actions, and $\mathcal{TA}$ with functionalities;

and

- the temporal existence of $\mathcal{DA}$ with validity time-intervals, $\mathcal{SA}$ with activity time-intervals, and $\mathcal{TA}$ with operation time-intervals.

3.2. Modeling steps

In the multi-type artifact framework, the modeling steps consist of connecting artifacts together. To this end, the framework resorts to Ghajargar et al.'s relations Ghajargar et al. (2018) to define situational relations between artifacts ($\mathcal{A}_i$) that would reside in a cyber-physical, social system. Ghajargar et al.'s modeling technique for IoT artifacts design and analysis refers to 4 relations namely, *augment-me* (Fig. 3), *comply-with-me*, *engage-me*, and *make-me-think* Ghajargar et al. (2018).

With respect to Fig. 2, we connect $\mathcal{DA}$, $\mathcal{TA}$, and $\mathcal{SA}$ together using our 4 situational relations that are *work-with-me*, *work-for-me*, *back-me*, and *avoid-me*. We discuss each situational relation in terms of rationale and components and then, exemplify it with particular artifact-types. Notations used in the discussion below are 0[...]n and 1[...]n standing for zero-to-many and one-to-many, respectively.

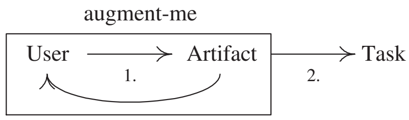

Figure 3: Representation of *augment-me* relation (Ghajargar et al. (2018))

work-with-me (Fig. 4): it has an initiator $\mathcal{A}_i$ and a set of partners $1[\mathcal{A}_j]n$ with whom the initiator collaborates ① when provisioning a complex service ② to a cyber-physical, social system ③. The collaboration between all artifacts remains active until the initiator artifact is stopped because of its time interval. In addition, all artifacts participating in *work-with-me* are accountable to the cyber-physical, social system. For illustration, a sensing $\mathcal{TA}$ would work with a $\mathcal{DA}$ that captures sensed temperatures to ensure that these temperatures are made available for a group of $\mathcal{TA}$ in a timely manner.

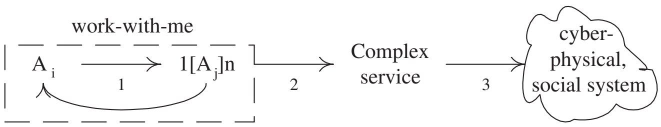

Figure 4: Representation of *work-with-me*

work-for-me (Fig. 5): it has an initiator $\mathcal{A}_i$ and a set of potential partners $0[\mathcal{A}_j]n$ (potential because partners might not always exist compared to *work-with-me* relation) that the initiator asks for assistance with provisioning a service ① to a cyber-physical, social system ②. The assistance between all artifacts remains active even if the initiator artifact is stopped because of its time interval. However, the assistance ends when the partner artifacts' time intervals expire. While all the partner artifacts participating in *work-for-me* are directly accountable to the initiator artifact, only the initiator artifact is accountable to the cyber-physical, social system. For illustration, an actuating $\mathcal{TA}$ could make a $\mathcal{TA}$ peer handle the pending operations of generating some necessary $\mathcal{DA}$.

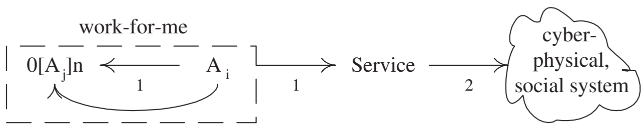

Figure 5: Representation of *work-for-me*

back-me (Fig. 6): it has an initiator $\mathcal{A}i$ and a set of potential partners $0[\mathcal{A}_j]n$ that the initiator would ask for assistance, should it cannot provision a service ① to a cyber-physical, social system ②. The assistance between all artifacts remains active until the partner artifacts are stopped when their time intervals expire. In term of accountability to a cyber-physical, social system, it shifts from the initiator artifact to the rest of partner artifacts. For illustration, a post $\mathcal{SA}$ would agree with a $\mathcal{SA}$ peer to relay its post to a group of $\mathcal{TA}$, should it stop working.

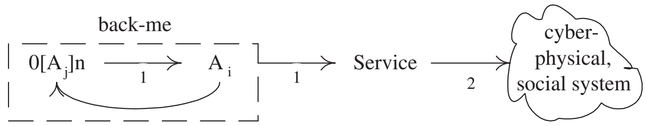

Figure 6: Representation of *back-me*

avoid-me (Fig. 7): it has an initiator $\mathcal{A}_i$ and a set of opponents $0[\mathcal{A}_j]n$ deemed undesirable by the initiator ① when provisioning a service ② to the cyber-physical, social system ③. For illustration, 2 sensing $\mathcal{SA}$ are not allowed to simultaneously sense the same surrounding to avoid conflicting details.

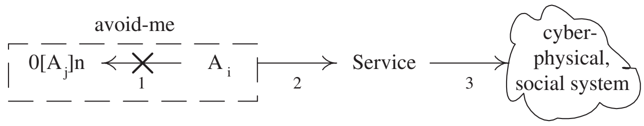

Figure 7: Representation of *avoid-me*

To wrap-up the discussions about situational relations, these relations either motivate artifacts to engage in collaborative scenarios or prevent artifacts from engaging in conflicting scenarios. Each scenario has specific requirements to satisfy and specific objectives to achieve. We also delineate the responsibilities of each artifact in term of accountability to the cyber-physical, social system. Artifacts' operation/activity time-intervals also impact how long the situational relations would remain active. This impact is detailed in the development steps following the synchronization of all artifacts' time intervals including validity intervals.

3.3. Development steps

In the multi-type artifact framework, the development steps target first, the coordination of social and thing artifacts participating in situational relations and then, the management of the data artifacts that these social and thing artifacts produce. To carry out the development steps, the framework resorts to Allen's time algebra Allen (1983) and relies on artifacts' time intervals so that potential (exclusive) time-relations between these artifacts are recommended. With respect to Allen's time algebra, there are 13 time relations like *precedes*, *equals*, *overlaps*, and *finishes*. However, not all of them would apply to our situational relations.

3.3.1. Combining time intervals

For illustration purposes, we use 2 situational relations, *work-with-me* and *avoid-me*, and $[b,e]$ notation to set an interval's *begin time* and *end time*.

***work-with-me*.** Considering this situational relation's characteristics (Fig. 4), the recommended time relations between a couple of artifacts, $\mathcal{A}_i$ (initiator) and $\mathcal{A}j$ (excluding $\mathcal{DA}$), participating in this relation would be:

1. *equals*($\mathcal{A}_i[b_i,e_i]$,$\mathcal{A}_j[b_j,e_j]$) where $b_i = b_j$ and $e_i = e_j$. Both $\mathcal{A}_i$ and $\mathcal{A}_j$ start together and end together making $\mathcal{A}_j$'s time interval completely available for $\mathcal{A}_i$. The combined time-interval that would ensure the successful completion of *work-with-me* would be $[b_i|b_j,e_i|e_j]$.
2. *starts*($\mathcal{A}_i[b_i,e_i]$,$\mathcal{A}_j[b_j,e_j)$]) where $b_i = b_j$. Both $\mathcal{A}_i$ and $\mathcal{A}_j$ start together but $\mathcal{A}_i$ ends before $\mathcal{A}_j$ requiring that $\mathcal{A}_i$ requests the collaboration of $\mathcal{A}_j$ before its time interval expires. As a result, the combined time-interval that would ensure the successful completion of *work-with-me* would be $[b_i|b_j,e_i]$.
3. *finishes*($\mathcal{A}_i[b_i,e_i]$,$\mathcal{A}_j[b_j,e_j)$]) where $e_i = e_j$. Both $\mathcal{A}_i$ and $\mathcal{A}_j$ end together but $\mathcal{A}_j$ starts first. When $\mathcal{A}_i$ requests the collaboration of $\mathcal{A}_j$, this assumes that $\mathcal{A}_j$ is available which is not always the case as $\mathcal{A}_j$ could be participating in other situational relations that where formed before $\mathcal{A}_i$'s situational relation. As a result, the combined time-interval that would ensure the successful completion of work-with-me would be $[b_i,e_i|e_j]$.
4. *overlaps*($\mathcal{A}_i[b_i,e_i]$,$\mathcal{A}_j[b_j,e_j)$]) where $b_i < b_j$ and $e_i > b_j$. Both $\mathcal{A}_i$ and $\mathcal{A}_j$ start separately and end separately requiring that $\mathcal{A}_i$ requests the collaboration of $\mathcal{A}_j$ before its time interval expires. As a result, the combined time-interval that would ensure the successful completion of *work-with-me* would be $[b_j,e_i]$.
5. *during*($\mathcal{A}_i[b_i,e_i]$,$\mathcal{A}_j[b_j,e_j)$]) where $b_i > b_j$ and $e_i < e_j$. Both $\mathcal{A}_i$ and $\mathcal{A}_j$ start separately and end separately. When $\mathcal{A}_i$ requests the collaboration of $\mathcal{A}_j$, a similar description to *finishes* time-relation could happen with regard to $\mathcal{A}_j$'s availability. As a result, the combined time-interval that would ensure the successful completion of *work-with-me* would be $[b_i,e_i]$.

By analyzing the 5 time relations above, the combined time-interval that would ensure the combined activeness of *work-with-me* across all the cases would have as begin time the latest begin-time among the 2 artifacts, and as end time the earliest end-time among the 2 artifacts.

We now proceed with analyzing the overlaps that would exist between a situational relation's combined time-interval and a data artifact's validity time-interval. These overlaps would again correspond to specific Allen's time intervals namely, *equals*, *overlaps*, *starts*, *finishes*, and *during*. For each time interval, we adjust the $\mathcal{C}$ombined time-interval ($[\mathcal{C}_b,\mathcal{C}_e]$) and $\mathcal{V}$alidity time-interval ($[\mathcal{V}_b,\mathcal{V}_e]$) resulting in a common time-interval. For illustration purposes, only, *during* time-relation is mentioned where *during*($[\mathcal{C}_b,\mathcal{C}_e]$,$[\mathcal{V}_b,\mathcal{V}_e]$) would have $[\mathcal{C}_b,\mathcal{C}_e]$ as a common time-interval.

***avoid-me*.** Considering this situational relation's characteristics (Fig. 7), the recommended time relations between a couple of artifacts, $\mathcal{A}_i$ (initiator) and $\mathcal{A}j$ (excluding $\mathcal{DA}$), participating in this relation would be:

1. *precedes*($\mathcal{A}_i[b_i,e_i]$,$\mathcal{A}_j[b_j,e_j]$) where $e_i - b_j < 0$. Both $\mathcal{A}_i$ and $\mathcal{A}_j$ have complete different time-intervals and hence, there is no need to define the combined time-interval that would ensure the successful completion of *avoid-me*. The success is already guaranteed.
2. *meets*($\mathcal{A}_i[b_i,e_i]$,$\mathcal{A}_j[b_j,e_j]$) where $e_i = b_j$. The analysis is similar to *precedes* situational relation.

Due to lack of a combined time-interval in the 2 recommended time relations above, there is no need to analyze any overlap between this interval and a data artifact's validity time-interval.

3.3.2. Binding artifacts to relations

For a successful binding of all thing, social, and data artifacts to situational relations' common time-intervals, the framework enforces this binding using a set of policies that define for instance, who the participants in a situational relation are, what actions these participants can/-cannot perform, and when these participants can pull out from a situational relation. The framework resorts to Ponder as a policy specification language Damianou et al. (2001) although other policy languages could be used without any impact on how the binding should occur

work-with-me characteristics include an initiator artifact $\mathcal{A}_i$, a partner artifact $\mathcal{A}_j$, a complex service to provision, an obligation of the partner artifact to collaborate with the initiator artifact on completing the complex service, and a refrain on the partner artifact from pulling out from the situational relation until the completion of the complex service.

1. First we capture the obligation with Listing 1 where line 1 refers to the obligation policy, line 2 triggers the policy following the acceptance of the partner artifact to take part in a *work-with-me* situational relation, line 3 refers to the partner artifact, line 4 refers to the initiator artifact, and, finally, line 5 refers to the sequence of actions forcing both artifacts to collaborate.
2. Second we capture the refrain with Listing 2 where line 1 refers to the refrain policy, line 2 refers to the partner artifact, line 3 refers to the initiator artifact, and line 4 refers to the action of withdrawing that the partner artifact should refrain from doing as long as the status of *work-with-me* situational relation is active as per line 5.

Listing 1: Obligation policy in support of *work-with-me*

```
1 inst oblig workwithmeObligPolicy {
2     on          work-with-me(''accept'',𝒜_j);
3     subject     s=𝒜_j;
4     target      t=𝒜_i;
5     do          t.invite(𝒜_i) → s.collaborate(𝒜_i);}
```

Listing 2: Refrain policy in support of *work-with-me*

```
1 inst refrain workwithmeRefrainPolicy {
2     subject     s=𝒜_j;
3     target      t=𝒜_i;
4     action      s.withdraw(𝒜_i);
5     when        work-with-me.status = ''active''}
```

avoid-me characteristics include an initiator artifact $\mathcal{A}_i$, an opponent artifact $\mathcal{A}_j$, a service to provision, and a refrain on the opponent artifact from residing in the initiator artifact's ecosystem until the completion of the service. We capture the refrain with Listing 3 where line 1 refers to the refrain policy, line 2 refers to the opponent artifact, line 3 refers to the initiator artifact, and line 4 refers to the action of interacting that the opponent artifact should refrain from doing as long as the status of *avoid-me* is active and the completion status of the initiator artifact's service is in-progress as per line 5.

Listing 3: Refrain policy in support of *avoid-me*

```
inst refrain avoidmeRefrainPolicy {
    subject     s=𝒜_j;
    target      t=𝒜_i;
    action      s.interact(𝒜_i);
    when        avoid-me.status = ''active'' ∧ t.service = ''in-progress''}
```

work-for-me characteristics include an initiator artifact $\mathcal{A}_i$, a partner artifact $\mathcal{A}_j$, a service to provision, and a request of assistance, in a form of a delegation, of the initiator artifact to the partner artifact to complete the service. We capture the assistance request with Listing 4 where line 1 refers to the delegation policy that implicitly hints to the initiator artifact, line 2 refers to the partner artifact, and line 3 refers to the action that the partner artifact is expected to perform on behalf of the initiator partner as long as this partner artifact's time interval is active as per line 4.

Listing 4: Delegation policy in support of *work-for-me*

```
1 inst deleg+ workformeDelegPolicy {
2     grantee     g = 𝒜_j;
3     action      process(service);
4     valid       g.time-interval = ''active'';}
```

After defining the necessary policies that enforce the participation of artifacts in situational relations, the final step is how to make these artifacts interact so they synchronize their actions. An interaction would refer to a message defined as a tuple ($< id, type, from, to, cnt >$) where *cnt* would be a content conveyed $from$ a *sender* $\mathcal{A}_i$ *to* a *receiver* $\mathcal{A}_j$. To identify the necessary message types with respect to the situational relations' characteristics, we draw some analogy with network protocols (e.g., Wondracek et al. (2008)) resulting in 5 message types namely, *open*, *sync*, *success*, *ack*, and *close*. In Table 4, we provide some examples of how the messages would be rolled out.

Table 4: Messages between artifacts participating in situational relations

Relation	Message				Description
	type	*from*	*to*	*cnt*	
work-with-me *equals*	*open*	$\mathcal{A}_i$	$\mathcal{A}_j$	$\mathcal{A}_i[b_i, e_i]$	informs $\mathcal{A}_j$ about $\mathcal{A}_i$'s start and end times
	ack	$\mathcal{A}_j$	$\mathcal{A}_i$	*null*	confirms $\mathcal{A}_j$ readiness
	sync	$\mathcal{A}_i$	$\mathcal{A}_j$	$*[data]n$	initiates $\mathcal{A}_i$,$\mathcal{A}_j$ collaboration
	ack	$\mathcal{A}_j$	$\mathcal{A}_i$	*success*	confirms $\mathcal{A}_j$ completion and hence, the collaboration
	close	$\mathcal{A}_i$	$\mathcal{A}_j$	*null*	ends the communication
avoid-me *precedes*	*open*	$\mathcal{A}_i$	$\mathcal{A}_j$	*suspension*	requests $\mathcal{A}_j$ to suspend its operation
	ack	$\mathcal{A}_j$	$\mathcal{A}_i$	*null*	confirms $\mathcal{A}_j$'s suspension
	close	$\mathcal{A}_i$	$\mathcal{A}_j$	*null*	ends the communication

4. Technical validation

In this section, we detail the technical development of the multi-type artifact framework along with the experiments that were performed to verify its technical doability and to evaluate its performance using a healthcare-driven IoT case-study. In this case study medical data are sensed/collected and then, transmitted to different artifacts.

4.1. Case study

Our healthcare-driven IoT case-study builds upon what is presented in Al-Khafajiy et al. (2019) and Maamar et al. (2019). Recent advances in ICT have facilitated the emergence of a new generation of IoT applications used in different contexts like smart cities and smart health. Juniper research [3] forecasts that by 2026 smart hospitals will deploy 7.4 million connected Internet of Medical Things (IoMT) devices. Also, according to Statista[4] the total number of wearable units shipped in 2022 reached 492 million. Finally, Gartner forecast[5] shows a growth of IoT spend by healthcare providers from $21 billion in 2019 to $54 billion in 2029 at a compound annual growth rate of 10%. IoT is simplifying user-2-machine-2-user interactions using sensor-enabled edge devices like smartphones and smart watches. IoT users now are able to monitor their health status, and control and regulate their equipment and appliances anywhere, anytime defining the cyber-physical space. Similarly, the social space is also evolving in the sense of allowing users to tweet/post/request data online to initiate desired services. This is expected to grow further with the emergence of metaverse, which will transform the healthcare industry towards more digitalized health services Tan et al. (2022).

The healthcare case-study adopted to validate the multi-type framework refers to a system for monitoring patients in care homes. The system's components are depicted in Fig. 2 allowing to monitor patients' vital signs via wearable sensors (forming the cyber-physical space) and to share details about these vital signs online (forming the social space) via social media platforms such as Twitter. As per Fig. 8, the cyber-physical space is associated with $\mathcal{TA}$ and the social space is associated with $\mathcal{SA}$. Both artifacts produce $\mathcal{DA}$ via the respective analyzers in Fig. 8. Furthermore, 2 executors, $\mathcal{S}$-Executor and $\mathcal{T}$-Executor, subscribe to an announcement platform where $\mathcal{DA}$ are posted for additional processing by relevant executors.

4.2. System in-operation

The system's simulator programmed in Python runs over 3 stages referred to as *configuration*, *execution*, and *monitoring*. The *configuration* stage has the following steps:

1. Deploy a set of virtual machines acting as devices in the simulator. Each device has separate processing, storage, and communication capabilities. These devices host $\mathcal{SA}$ and/or $\mathcal{TA}$ considering *work-with-me*, *work-for-me*, *back-me*, and *avoid-me* situational relations. It has been assumed that these relations are equally distributed for $\mathcal{TA}$ (having $\mathsf{R}_{i=1,4}$=0.25). However, *back-me* and *avoid-me* for $\mathcal{SA}$ have been dominant with approximately 75%.

2. Organize devices into groups where a group is associated with a specific domain. As per the current case study, the domains are applications from the healthcare industry such as domain_1=temp-sensor$_{care_1}$ and domain_2=ingestible-sensor$_{care_2}$.

3. Generate a random number (*nb*) of $\mathcal{SA}$ and $\mathcal{TA}$ per domain. This number is constrained by an interval of 2 values ($[min, max]$). Afterwards, the obtained $\mathcal{SA}$ and $\mathcal{TA}$ generate $\mathcal{DA}$. S-$\mathcal{DA}$ are based on tweets collected from a real dataset Lamsal (2019) and processed by the $\mathcal{S}$-Analyzer (Fig. 8).

---

[3] www.juniperresearch.com/press/smart-hospitals-to-deploy-over-7mn-iomt.

[4] www.statista.com/statistics/437871/wearables-worldwide-shipments.

[5] www.gartner.com/en/documents/3990045.

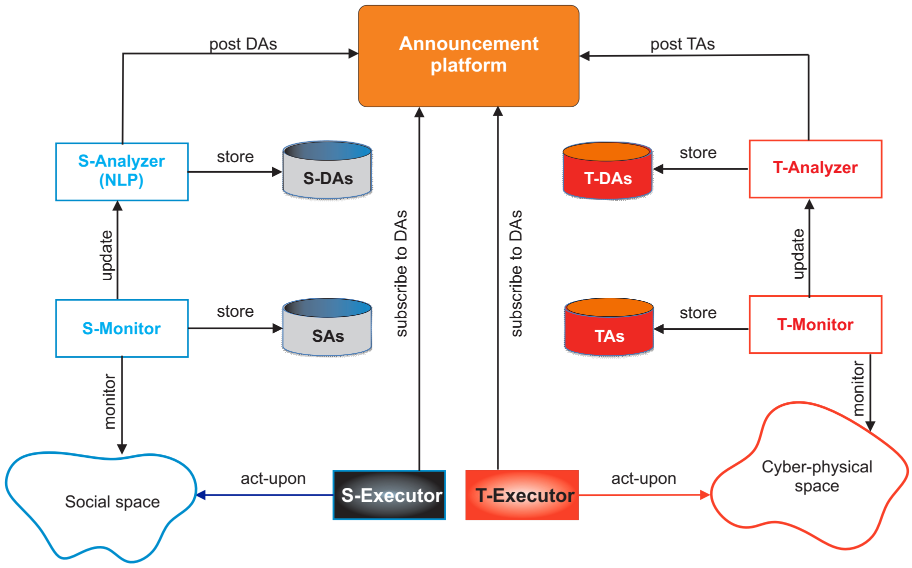

Figure 8: System architecture of the multi-type artifact framework

4. Set execution and deadline times for the generated $\mathcal{SA}$ and $\mathcal{TA}$ based on their respective creation times. Similarly, we associate $\mathcal{DA}$ with a Processing Time (PT) defining how long it will lock a device for deployment whether it is a request for more data from a sensor or a tweet posted on Twitter by a patient.

5. Finally, set a probability of having relation types between $\mathcal{SA}$ and $\mathcal{TA}$ occur, so that we control the relations within the simulation to perform different tests.

In the *execution* stage, the simulator proceeds with every newly-established $\mathcal{TA}$- and $\mathcal{SA}$-relation obtained in step 1 of the configuration stage as follows:

1. Identify a time-relation that corresponds to the selected situational relation in order to determine the artifact(s) that will start first.

2. Deploy and execute the artifact(s) participating in the identified time-relation based on their respective begin-times over devices. It could happen that some devices do not have enough resources. In this case, the deployment is put on-hold until some devices become available after other artifact(s) either complete their works or are discarded because they do not participate in any relation.

Finally, the *monitoring* stage runs in conjunction with the *execution stage* allowing the simulator to "keep an eye" on the amount of available resources per device and to track the execution progress of all artifacts, $\mathcal{TA}$, $\mathcal{SA}$, and $\mathcal{DA}$. To this end, the simulator proceeds as follows:

1. Track the events that could impact a device's load by checking whether any artifact execution was recently launched over this device, any artifact execution was either completed or re-scheduled with the remaining time to complete, or any artifact execution was resumed (i.e., load increases).

2. Track the events that could impact artifact execution by checking whether any artifact deployment was put on-hold due to unavailable resources or any artifact's deadline expired before the work is done (i.e., $\mathcal{TA}$ failure).

4.3. Experiments

The experiments are to study how the number of different multi-type artifacts would impact the response time of the healthcare monitoring system.

The first experiment examines the impact of processing $\mathcal{TA}$ and $\mathcal{SA}$ according to R (configuration stage's step 1). This will reveal how long $\mathcal{TA}$ or $\mathcal{SA}$ would wait before they get executed according to the types of situational relations. To ensure the experiment's representativeness, we iterate it 100 times. Fig. 9 shows the average waiting-time of $\mathcal{TA}$ or $\mathcal{SA}$. It is clear that $\mathcal{SA}$ have lowest waiting times compared to $\mathcal{TA}$ and this can be due to the types of situational relations. It can be explained by the fact that situational relations between $\mathcal{TA}$ may have constraints due to *work-with-me* and *work-for-me* relations that require one $\mathcal{TA}$ to wait for another $\mathcal{TA}$ be active according to some time relations (e.g., *equals*, *starts*, and *finishes* as per Section 3.2).

The second experiment adopts the first experiment's settings, but having different number of $\mathcal{TA}$ and $\mathcal{SA}$ at each iteration. We refer to this as artifact-type density in the simulation. It starts by 30% $\mathcal{SA}$ and 70% $\mathcal{TA}$, then we gradually increase the number of $\mathcal{TA}$ by 10% until we got to the opposite distribution of $\mathcal{SA}$ and $\mathcal{TA}$ (i.e., 70% $\mathcal{SA}$ and 30% $\mathcal{TA}$) as per Fig. 10. It is now evident that even with more $\mathcal{SA}$ the average waiting time is less than what $\mathcal{TA}$ requires.

The third experiment is for the scalability evaluation. Here we run the simulator with over 1K artifacts. The distribution of types of artifacts is equal in average. The results obtained from the new configurations have shown a reasonably scalable framework. In Fig. 11 a box plot presents the mean waiting-time (red dash inside the box), a box top is the max waiting-time and a box bottom is the lowest-waiting-time for the number of artifacts in each iteration. This was important to test the scalability of the multi-type artifact framework, which is obviously proven to be scalable given the linear increase in waiting time.

5. Future work

In Section 3.2, we refer to Ghajargar et al.'s *augment-me*, *comply-with-me*, *engage-me*, and *make-me-think* relations that basically connect users to IoT artifacts, tasks that users initiate, and situations in which users reside, together Ghajargar et al. (2018). Along with these relations, we suggested our *work-with-me*, *work-for-me*, *back-me*, and *avoid-me* situational relations that connect artifacts together. While we label Ghajargar et al.'s relations and ours relations as $\mathcal{P}2\mathcal{A}$ and $\mathcal{A}2\mathcal{A}$ standing, respectively, for $\mathcal{P}$erson-2-$\mathcal{A}$rtifact and $\mathcal{A}$rtifact-2-$\mathcal{A}$rtifact, we would like to consider additional relations that we would label as $\mathcal{A}$rtifact-2-$\mathcal{P}$erson ($\mathcal{A}2\mathcal{P}$) allowing to close the loop of interactions that a person would initiate with things.

Fig. 12 illustrates the cycle of enabling all relations between persons and artifacts where $\mathcal{P}2\mathcal{A}$ relations are the entry point to the cycle followed by $\mathcal{A}2\mathcal{A}$ relations then, $\mathcal{A}2\mathcal{P}$ relations. Due to the nature of $\mathcal{A}2\mathcal{P}$ relations, we would specialize them into *notify-her* so that the artifact informs the person about any updates, *monitor-her* so that the artifact controls what the person is doing, and *allow-her* so that the artifact makes the person accomplish something specific. In term of future work we would like to specify the new relations like we did with the situational relations as well as ensure the consistency between all relations to avoid conflicting situations.

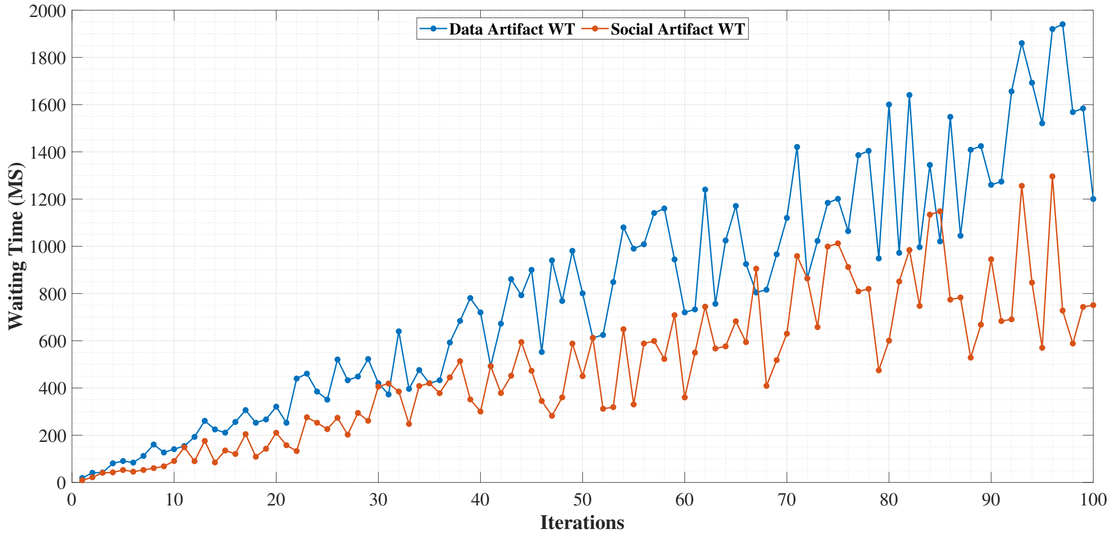

Figure 9: TAs' and SAs' average waiting-times

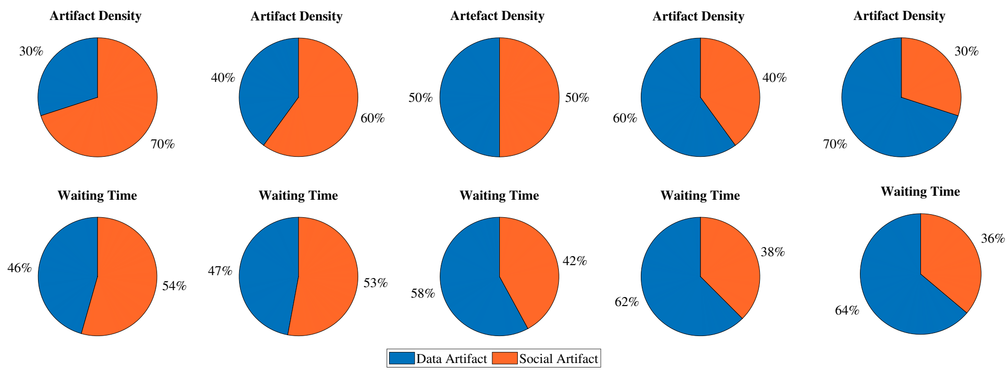

Figure 10: Artifact-type density impact over average waiting time

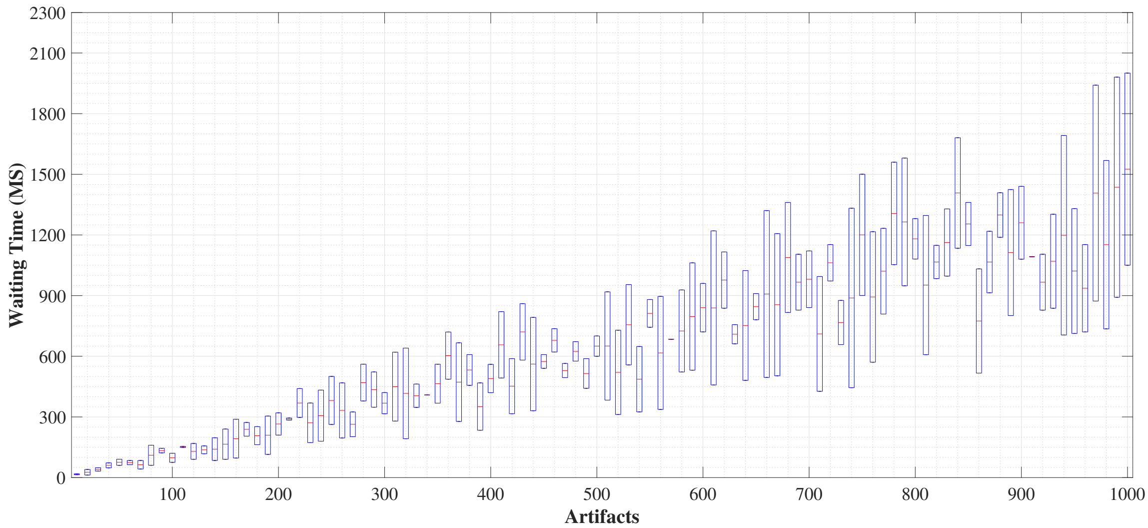

Figure 11: Scalability per number of artifacts

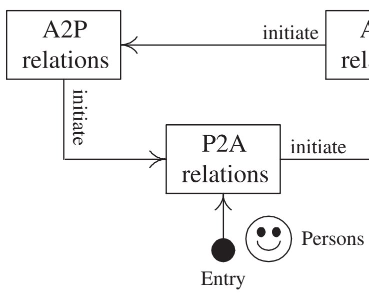

Figure 12: Interaction-loop closure between persons and artifacts

6. Conclusion

This paper presented a framework for the design and development of cyber-physical, social systems. This framework's characteristics include artifacts specialized into data, social, and thing, situational relations specialized into work-for-me, work-with-me, back-me, and avoid-me, and policies enforcing the binding of artifacts to theses relations. The policies are specified in Ponder and took into account time validity/activity/operation intervals confirming the availability of artifacts. This confirmation is subject to identifying potential Allen's time-relations between the different intervals. The framework was technically validated and evaluated based on a healthcare-driven IoT case-study in which medical data were sensed/collected and then, transmitted to relevant artifacts. In term of future work, we would like to examine the appropriateness of defining new relations between persons and artifacts allowing in fact to close the interaction loop between all the framework's 2 main stakeholders namely, persons and artifacts.

References

Al-Khafajiy, M., Baker, T., Chalmers, C., Asim, M., Kolivand, H., Fahim, M., Waraich, A., 2019. Remote Health Monitoring of Elderly through Wearable Sensors. Multimedia Tools and Applications 78.

Allen, J., 1983. Maintaining Knowledge about Temporal Intervals. Communications of the ACM 26.

Badr, Y., Maamar, Z., October 2009. Can Enterprises Capitalize on Their Social Networks? Cutter IT Journal 22.

Baxter, G., Sommerville, I., 2011. Socio-Technical Systems: From Design Methods to Systems Engineering. Interacting With Computers 23.

Burégio, V.A., Kajan, E., Sellami, M., Faci, N., Maamar, Z., Benslimane, D., 2016. Revisiting Software Engineering in the Social Era. International Journal of Systems and Service-Oriented Engineering 6.

Chourabi, H., Nam, T., Walker, S., Gil-García, J., Mellouli, S., Nahon, K., Pardo, T., Jochen Scholl, H., 2012. Understanding Smart Cities: An Integrative Framework, in: Proceedings of the 45th Hawaii International International Conference on Systems Science (HICSS'2012), Grand Wailea, Maui, USA.

Damianou, N., Dulay, N., Lupu, E., Sloman, M., 2001. The Ponder Policy Specification Language, in: Proceedings of the International Workshop on Policies for Distributed Systems and Networks (POLICY'2001), Bristol, UK.

DZone, 2017 (visited in May 2017). The Internet of Things, Application, Protocols, and Best Practices. Technical Report. https://dzone.com/guides/iot-applications-protocols-and-best-practices.

Emmanouilidis, C., Pistofidis, P., Bertoncelj, L., Katsouros, V., Fournaris, A., Koulamas, C., Ruiz-Carcel, C., 2019. Enabling the Human in the Loop: Linked Data and Knowledge in Industrial Cyber-Physical Systems. Annual Reviews in Control 47.

Friedow, C., Völker, M., Hewelt, M., 2018. Integrating IoT Devices into Business Processes, in: Proceedings of the 1st Workshop on Flexible Advanced Information Systems (FAiSE'2018) held in conjunction with CAiSE'2018, Tallinn, Estonia.

Ghajargar, M., Wiberg, M., Stolterma, E., 2018. Designing IoT Systems that Support Reflective Thinking: A Relational Approach. International Journal of Design 12.

Hehenberger, P., Vogel-Heuser, B., Bradley, D., Eynard, B., Tomiyama, T., Achiche, S., 2016. Design, Modelling, Simulation and Integration of Cyber Physical Systems: Methods and Applications. Computers in Industry 82.

Jakubowska, L., 2011. Methods and Techniques of Cyberspace Research Theory and Practice.

Kajan, E., Faci, N., Maamar, Z., Sellami, M., Ugljanin, E., Kheddouci, H., Stojanovic, D., Benslimane, D., 2020. Real-time tracking and mining of users' actions over social media. Comput. Sci. Inf. Syst. 17, 403–426.

Kumaran, S., Liu, R., Wu, F.Y., 2008. On the Duality of Information-Centric and Activity-Centric Models of Business Processes, in: Proceedings of the 20th International Conference on Advanced Information Systems Engineering (CAiSE'2008), Montpellier, France.

Lamsal, R., 2019. Twitter Sentiment Analysis Data. URL: `https://dx.doi.org/10.21227/t4mp-ce93`, doi:`10.21227/t4mp-ce93`.

Law, J., Hassard, J., 1999. Actor Network Theory and After. Blackwell Publishers.

Lee, E., 2008. Cyber Physical Systems: Design Challenges, in: Proceedings of the 11th IEEE International Symposium on Object and Component-oriented Real-Time Distributed Computing (ISORC'2008), Vasteras, Sweden.

Maamar, Z., 2022. Thing Artifacts: Definitions, Challenges, and Opportunities. Technical Report. Zayed University.

Maamar, Z., Badr, Y., Narendra, N.C., 2010. Business Artifacts Discovery and Modeling, in: Proceedings of the 8th International Conference on Service Oriented Computing (ICSOC'2010), San Francisco, California, USA.

Maamar, Z., Baker, T., Faci, N., Ugljanin, E., Al-Khafajiy, M., Burégio, V., 2019. Towards a Seamless Coordination of Cloud and Fog: Illustration through the Internet-of-Things, in: Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing (SAC'2019), Limassol, Cyprus.

Maamar, Z., Burégio, V., Sellami, S., Souto Rosa, N., Peng, Z., Subin, Z., Prakash, N., Benslimane, D., Silva, R., 2017. Bridging the Gap Between the Business and Social Worlds: A Data Artifact-Driven Approach. T. Large-Scale Data- and Knowledge-Centered Systems 35, 27–49.

Maamar, Z., Kajan, E., Guidara, I., Moctar-M'Baba, L., Sellami, M., 2020. Bridging the Gap between Business Processes and IoT, in: Proceedings of the 24th International Symposium on Database Engineering & Applications (IDEAS'2020), Seoul, Republic of Korea.

Mitchell, W., 2007. Intelligent Cities. http://www.uoc.edu/uocpapers/5/dt/eng/mitchell.pdf.

Narendra, N.C., Badr, Y., Thiran, P., Maamar, Z., 2009. Towards a Unified Approach for Business Process Modeling Using Context-Based Artifacts and Web Services, in: Proceedings of the 2009 IEEE International Conference on Services Computing (SCC'2009), Bangalore, India.

Nass, C., Moon, Y., 2000. Machines and Mindlessness: Social Responses to Computers. Journal of Social Issues 56.

Nigam, A., Caswell, N., 2003. Business Artifacts: An Approach to Operational Specification. IBM Systems Journal 42, 428–445.

Popova, V., Fahland, D., Dumas, M., 2015. Artifact Lifecycle Discovery. International Journal of Cooperative Information Systems 24.

Rejeb, A., Rejeb, K., Treiblmaier, H., Appolloni, A., Alghamdi, S., Alhasawi, Y., Iranmanesh, M., 2023. The Internet of Things (IoT) in Healthcare: Taking Stock and Moving Forward. Internet of Things 22, 100721.

Schneckenberg, S., 2009. Web 2.0 and the Empowerment of the Knowledge Worker. Journal of Knowedge Management 13.

Tan, T., Li, Y., Lim, J., Gunasekeran, D., Teo, Z., Ng, W., Ting, D., 2022. Metaverse and Virtual Health Care in Ophthalmology: Opportunities and Challenges. The Asia-Pacific Journal of Ophthalmology 11.

Törngren, M., Grogan, P., 2018. How to Deal with the Complexity of Future Cyber-Physical Systems? Designs 2, 40.

Turban, E., Bolloju, N., Liang, T., 2011. Enterprise social networking: Opportunities, adoption, and risk mitigation. J. Org. Computing and E. Commerce 21, 202–220. URL: `https://doi.org/10.1080/10919392.2011.590109`, doi:`10.1080/10919392.2011.590109`.

Wayne, H., 2007. The Good News and the Bad News. Computer 40.

Wondracek, G., Comparetti, P.M., Kruegel, C., Kirda, E., 2008. Automatic Network Protocol Analysis, in: Proceedings of the 15th Annual Network and Distributed System Security Symposium (NDSS'2008), San Diego, CA, USA.

Yao, X., Zhou, J., Lin, Y., Li, Y., Yu, H., Liu, Y., 2019. Smart Manufacturing based on Cyber-Physical Systems and Beyond. Journal of Intelligent Manufacturing 30.

Yilma, B., Panetto, H., Naudet, Y., 2021. Systemic Formalisation of Cyber-Physical-Social System (CPSS): A Systematic Literature Review. Computers in Industry 129.

Zeng, J., Yang, L., Lin, M., Ning, H., Ma, J., 2020. A Survey: Cyber-Physical-Social Systems and their System-level Design Methodology. Future Generation Computer Systems 105.