

Requirements Engineering for Cyber Physical Production Systems

Pericles Loucopoulos¹[0000-0003-1395-5001], Evangelia Kavakli²[0000-0003-2743-5146]
and Natalia Chechina³[0000-0001-6763-6052]

¹ Institute of Digital Innovation and Research
periloucopoulos@me.com

² University of the Aegean, Mytilene, Greece
kavakli@aegean.gr

³ Bournemouth University, Department of Computing and Informatics, UK
nchechina@bournemouth.ac.uk

Abstract. Traditional manufacturing and production systems are in the throes of a digital transformation. By blending the real and virtual production worlds, it is now possible to connect all parts of the production process: devices, products, processes, systems and people, in an informational ecosystem. This paper examines the underpinning issues that characterise the challenges for transforming traditional manufacturing to a Cyber Physical Production System. Such a transformation constitutes a major endeavour for requirements engineers who need to identify, specify and analyse the effects that a multitude of assets need to be transformed towards a network of collaborating devices, information sources, and human actors. The paper reports on the e-CORE approach which is a systematic, analytical and traceable approach to Requirements Engineering and demonstrates its utility using an industrial-size application. It also considers the effect of Cyber Physical Production Systems on future approaches to requirements in dealing with the dynamic nature of such systems.

Keywords: Requirements Engineering, Industry 4.0, Factories of the Future (FoF), Cyber Physical Production Systems (CPPS), Capability-Oriented Modelling.

1 Introduction

A term that has come to represent the blending of the physical with the digital towards a new industrial era is that of Industry 4.0 [1]. More specifically to production systems, terms that have come to dominate the debate are those of Factory of the Future (FoF) and Cyber Physical Production Systems (CPPS) [2]. The purpose of CPPS is to transform current Physical Production Systems (PPS) towards those that, underpinned by ICT-related capabilities, could achieve a greater degree of connectivity of production entities and processes.

The aforementioned transformation could have a profound impact on operational, tactical and strategic aspects of a manufacturing company. *Operations* would be di-

rectly impacted on a daily basis through the augmentation of traditional physical production machines with devices capable of providing valuable information on the functioning of these machines. *Tactical* decisions could result in more efficient, streamlined production lines through the provision of data analyses for optimized planning, improved scheduling, and reactive as well as predictive event handling. *Strategically*, the ‘smart factory’ could be transformed into a profitable innovation centre through its ability to be quickly ‘reprogrammed’ to provide faster time-to-market responses to global consumer demand, effectively addressing mass-customization needs and bringing life to innovative new products.

The *challenge* to designers of systems aimed at facilitating this type of organisational transformation is that “... the boundaries between the real world and the virtual world become increasingly blurred” [3] with the resulting hybrid systems being regarded as “... online networks of social machines that are organised in a similar way to social networks” [1]. In such a setting, all *assets* of the manufacturing enterprise, whether human, physical or digital, need to be considered synergistically if the transformation is to be successful.

The *motivation* for the work presented in this paper is threefold. First, to present a methodological approach that addresses the aforementioned challenge, especially how an asset-centric view is utilised for a robust, analytical and traceable way for requirements for CPPS. This is done by introducing the e-CORE (early Capability Oriented Requirements Engineering) framework in terms of its conceptual foundations and its way of working. Second, to demonstrate the utility of this approach in the context of a real manufacturing situation. This is done through a detailed walkthrough of the way the approach was applied to two major transformational needs of a major automobile manufacturer, henceforth referred to as FCA, and which was done in the context of the Horizon 2020 project DISRUPT. Third, to present a reflective discussion on the way that e-CORE could facilitate the definition of self-awareness requirements for dealing with the interplay between system and its environment.

The paper is organised as follows. Section 2 sets the context of Physical Production Systems (PPS) in a general sense, in terms of the key concepts involved in a manufacturing production topology, and which will be influenced when incorporating a cyber-centric production chain. The e-CORE approach, is introduced in section 3, together with the support arguments for the reason to deploy a capability-orientated method to transforming PPS to CPPS. Section 4 demonstrates the complete e-CORE lifecycle in the FCA case noting that due to space limitation the conceptual models presented in this paper represent a small but nevertheless fully representative part of the actual results from this industrial case. Finally, section 5 concludes this paper with a short commentary on the value of the approach to CPPS applications and defines research initiatives for extending the approach towards dealing also with emergent system behaviours.

2 Challenges in the transformation towards CPPS

The term “Smart Factory” or “Factory of the Future” is used to define a flexible factory that can be dynamically transformed to respond to various events emanating from disruptions in the supply chain or in the production line. To this end, *production topologies* need to achieve dynamic re-configurability and scaling, to facilitate decision support by accommodating value chain collaboration and to enable decentralised (self-) adjustment of production, while also incorporating resource optimisation.

Production topologies define the context within which a transformation in manufacturing is to take place. Production topologies, according to the Instrument Society of America [4, 5], the taxonomic analysis of manufacturing systems [6-8], and ontologies for manufacturing and logistics [9], can be considered in terms of concepts that fall into two broad categories: *external* (e.g. SUPPLIER, TRANSPORTER and ORDER) and *internal* (e.g. MATERIAL, PERSONNEL, PRODUCTION LINE, WORK CELL, and EQUIPMENT).

The challenges to be considered by requirements engineers are of a technical, organisational and social nature such as: “what are the assets, human, physical, cyber, whose unique combination can provide strategic advantage to the enterprise?”; “what is the gap between existing and desired capabilities?”; “what is the requirements process such that one can reason about the choices available and the choices made in transforming existing assets to cyber-oriented ones?”. These challenges are addressed by the e-CORE approach which is a systematic way of proceeding from high-level early requirements to late requirements.

3 The e-CORE Approach

3.1 Background and related work

The requirements of new complex, emergent systems, such as smart factories, intelligent transportation, and smart cities, has brought new challenges to the RE discipline [10, 11]. These complex, heterogeneous systems of systems, consist of digital structures (e.g., software, data) that interact with physical constructs (e.g., sensors, devices) and with social elements (e.g., persons, organisations). The interplay between the different worlds of physical, digital and social has thus become more intricate, complex, dynamic, and generative. The challenges presented by production systems’ transformation towards cyber-physicality requires a new metaphor [12]. We posit that the notion of *capability* represents a most suitable metaphor that provides the means of considering the intertwining of concerns in these different worlds in a way such that it is possible to connect strategic objectives and high-level organizational requirements to technological artefacts in a unified manner.

The general consensus is that an enterprise capability represents a conceptual service that a group of processes and people, supported by the relevant application, information and underlying technology, performs [13, 14]. There has been a number of attempts at incorporating the notion of ‘capability’ in frameworks, most notably those of the Open Management Group [15], the Department of Defence Architecture Framework

(DOFAF) [16] and the NATO Architecture Framework (NAF) [17] as well as considering capability as a way of developing organisational strategies [18, 19]. In the context of information systems engineering, capability has been examined from a software development and management perspective, most notably in the Capability as a Service (CaaS) project [20].

In a capability-oriented paradigm we are interested in identifying possession of *valuable, rare, inimitable* and *non-substitutable* resources of enterprise as a source of sustainable advantage [21], whether these are existing capabilities or new ones that need to be introduced. Using capabilities as the starting point one can begin investigating and analysing what lies behind these fundamental enterprise assets, what goals govern them, what actors are involved and how they collaborate to synergistically meet requirements for enterprise transformation.

3.2 The e-CORE conceptual framework

The *conceptual modelling* framework underpinning e-CORE has been developed and applied in recent work [22-25]. It employs a set of complimentary and intertwined modelling paradigms based on enterprise *capabilities, goals, actors, and information objects*.

In CPPS applications, a key consideration is to focus on physical assets and in the way that these are transformed into on-line networks of collaborating social machines in a similar way to social networks [1]. In this context the e-CORE approach is particularly suitable since it focuses attention on assets and their collaboration for achieving a certain enterprise goal. This notion is shown in the meta-model of Fig. 1.

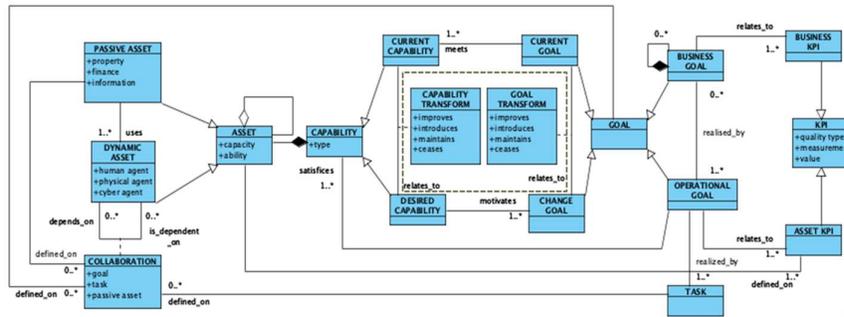


Fig. 1. The e-CORE top level meta-model

Referring to the meta-model in Fig. 1, one can see that a CAPABILITY is a generalisation of ASSETS (capacities and abilities) where ASSETS are distinguished between PASSIVE (resources) and DYNAMIC (agents). DYNAMIC ASSETS represent the social dimension, focusing on the COLLABORATION between these agents, which is defined as dependencies between them. These dependencies may involve the exchange of PASSIVE ASSETS (resources), or on execution of some TASK or the achievement of a GOAL.

In a RE setting we are interested in both `CURRENT CAPABILITIES` and `DESIRED CAPABILITIES` in order to model the necessary transformations from the former to the latter. There is a symmetry between `CURRENT CAPABILITIES` and `DESIRED CAPABILITIES` in the sense that each set is related to enterprise goals, the former to `CURRENT GOALS` and the latter to `CHANGE GOALS`. In e-CORE, requirements are modelled and analysed in terms of the juxtaposition of `CHANGE GOALS` against `CURRENT GOALS` and their corresponding capabilities.

3.3 The way-of-working

The e-CORE process and detailed activities are summarized in the three phases, as shown in Fig. 2.

Information elicitation refers to the collection of information related to the user case using a number of instruments (online forms, structures elicitation forms, collaborative workshops, onsite visits). It results in `user narratives` of the existing enterprise situation, as well as the user needs and aspirations with respect to the foreseen functionality and quality of the new system under development. The use of natural language in these user narratives has the advantage of ease of transferability but is prone to ambiguity and inconsistencies.

Defining those concepts that are relevant to the CPPS in a clear and consistent manner is done through `e-CORE concept identification`, which results in the list of concepts that describe the application domain. However, it does not define their structure, nor the interrelationships that exist between these concepts which in turn hinders any potential analysis.

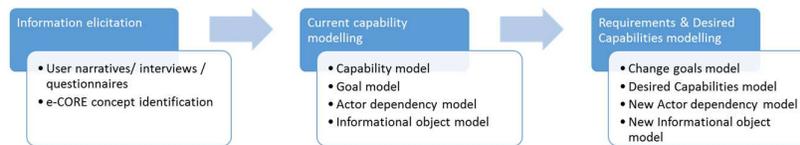


Fig. 2. The e-CORE Process

This is ameliorated by the second phase which is that of *current capability modelling*, according to the e-CORE meta-model, as discussed in section 3.2. In particular, 4 different types of modelling are considered each focusing on specific conceptual perspectives:

- The `capability model` focuses on the capacities and abilities necessary for a particular application.
- The `goal model` focuses on enterprise's objectives for retaining, acquiring or developing the necessary capabilities for the application.
- The `actor-dependency model` focuses on the socio-technical components of the enterprise that relate to specific capabilities for the application.
- The `informational model` focuses on the logical structure of the informational resources that are part of the enterprise's capabilities and act as the medium of communication between enterprise actors when they are trying to meet specific enterprise goals.

This model-driven approach encourages modellers to focus on those elements that are deemed to be key drivers in the dynamic change of enterprises and their systems, whilst ensuring that consistency is achieved across all four, through appropriate relationships between model concepts as defined in the e-CORE meta-model.

The third phase, *requirements and desired capabilities modelling*, focuses on the business requirements for change and the way these are mapped onto new capabilities and in the way these capabilities are operationalised in terms of actor dependencies. `Change goals` provide a way of identifying and reasoning about the user requirements and as such they express a desired state the user wishes to achieve. Reasoning about change goals is based on the premise that the desired changes are derived through the comparison of the ‘desired’ vision against the ‘present’ reality (the current goal model). This process aims to re-interpret each change requirement in relation to the existing goals, involving the key business stakeholders, most of whom would have been involved in the definition of current goals. The result of this activity is the construction of a revised goal model (the `Change Goal Model`) detailing stakeholders’ requirements for change. In the standard goal decomposition manner, the change goal model shows operational goals for the new improved situation. These operational goals, according to the e-CORE meta-model (**Fig. 1**), motivate `new capabilities` in terms of the new assets which may be further elaborated in the `desired actor dependency model`. In detailing the actor dependency model, it is then possible to proceed with the detailed functional and non-functional requirements of the desired system and its components.

4 Application of the e-CORE framework on the FCA case

FCA being one of the largest automotive manufacturers in the world has automated its production planning and scheduling according to the capabilities of the current software configuration. However, FCA still faces problems when there is a disruption in either its external topology, e.g. the *supply chain* (delays, errors in the supply of components) or in its internal topology, e.g. the *production process* (machine breakdown, unscheduled maintenance, software problems). The e-CORE approach has been used in collaboration between requirements engineers and a variety of FCA stakeholders (production planners, production engineers, and logistics teams). Sections 4.1-4.3 provide details of all 3 e-CORE process phases applied on the FCA case.

4.1 Information elicitation

Initially FCA requirements are expressed in natural language by the users, as shown in **Fig. 3**.

The narrative reveals a number of FCA business capabilities that can be informally mapped onto e-CORE, detailed in terms of the assets (in the form of human, physical and software actors) that the company possesses and the ability, (in the form of means or skills), inherent in these assets. For example, ‘`Production Scheduling`’ deploys

the 'Production Planner' (referred to as Valentina in the narrative) in collaboration with the software applications of 'OSS', 'MES' and 'MRP', having the knowledge, expertise and software ability for 'Scheduling' and 'Rescheduling'.

The screenshot shows a software interface titled "User Interview on Production Scheduling". The main text area contains a narrative about Valentina's role in production scheduling, mentioning tools like OSS, MES, and MRP, and tasks like re-planning and sequence computation. Below the text is a table of candidate classes identified from the narrative.

No.	Candidate Class	Extracted Text	Type	Description
1	OSS	OSS	Actor	Software tool for computing Job Sequence (Capacity)
2	Production Scheduling	production scheduling	Term	Capacity
3	Maintenance Team	colleagues from Manufacturing, Logistics, Quality or Maintenance	Actor	Human Actor (Capacity)
4	Rescheduling	re-plan (up to several times per day) the sequence of vehicles	Term	Ability
5	MES	MES	Actor	Software tool (Capacity)
6	MRP	MRP	Actor	Software tool (Capacity)
7	Scheduling	Order slotting and sequencing	Term	Ability
8	Production Planner	Valentina	Actor	Human Actor (Capacity)
9	Line configuration	sequence is computed	Term	Ability
10	Production Line	plant availability and capacity	Term	Capacity

Fig. 3. e-CORE concept identification from user narrative

4.2 Current capability modelling

Starting from the concepts that were identified in section 4.1, we can construct the FCA current capability model shown in Fig. 4, which defines 5 main capabilities denoted as 'CAP1: Inbound Logistics', 'CAP2: Material Management', 'CAP3: Production Planning', 'CAP4: Production Scheduling' and 'CAP5: Production'.

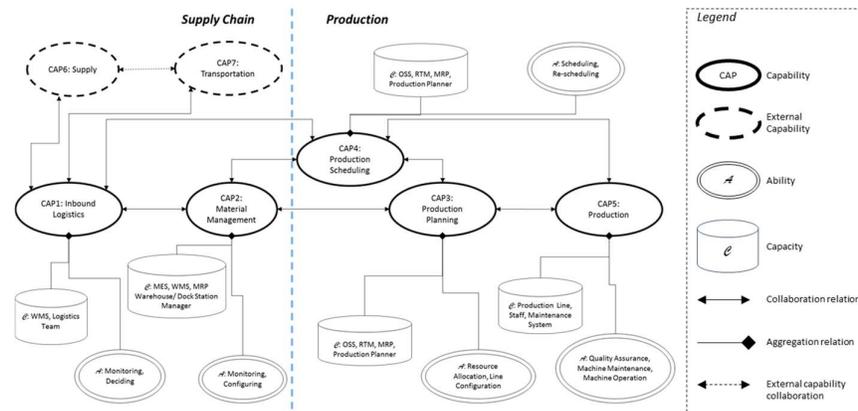


Fig. 4. The FCA Current Capability Model

The model of **Fig. 4** provides the scoping for the FCA application. 'CAP1' and 'CAP2' relate to the first problem area of supply of materials whilst 'CAP3', 'CAP4' and 'CAP5' relate to the second namely that of the production process.

In addition to these internal capabilities, there are two external capabilities that are owned by enterprises with which FCA collaborate but whose capabilities are not owned, controlled or subject to any influence by FCA. These are capabilities 'CAP6: Supply' and 'CAP7: Transportation'. They are included in the capability model in order to externalize these relationships, which may be very significant if in the transformed situation there may be opportunities for a closer collaboration of FCA with external entities for example suppliers and logistics companies, by making use of Internet of Things (IoT) functionalities.

The capability model acts as an *anchor point* for the rest of the e-CORE models, which for reasons of brevity are not presented here individually, but segments of these are shown in **Fig. 5** which also provides a visual representation of how these models, driven by the capability model are formally interrelated. These interrelationships objectively provide answers to the following questions: “*why does the enterprise need these capabilities?*” (answered by the *goal model*), “*what socio-technical actors are involved and how do they co-operate in order to meet these enterprise goals?*” (answered by the *actor dependency model*), and “*what kind of information is used in this co-operation?*” (answered by the *informational object model*).

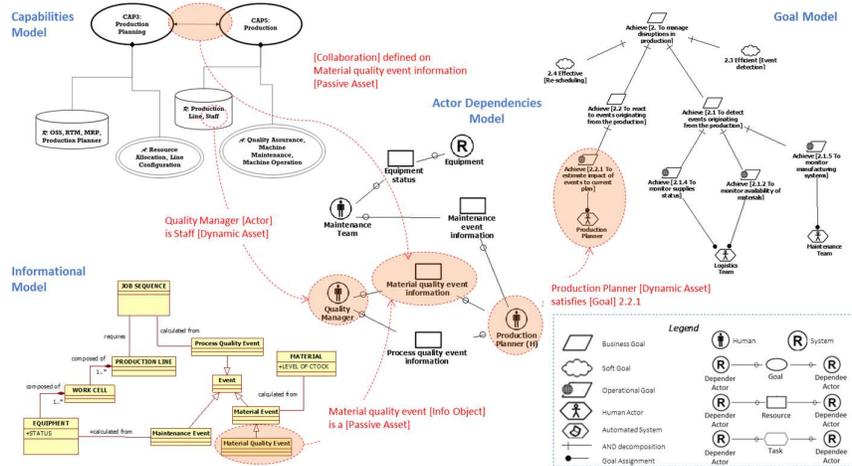


Fig. 5. Cross model relationships

As shown in **Fig. 5**, the dynamic asset of 'Staff' of capability 'CAP5' motivates the analysis in the actor-dependency model of this asset, which resulted in the actors 'Quality Manager' and 'Maintenance Team'.

The collaboration between capabilities 'CAP3' and 'CAP5', gave rise to the 'Material quality event information' exchange of resource between 'Quality

Manager' and 'Production Planner'. This resource is then identified as an informational resource, which was modelled in the informational model as 'Material Quality Event'.

Finally, the existence of 'Production Planner' in the actor-dependency model is due to the enterprise goal of 'G2.2.1. To estimate impact of to current plan' which is met by 'CAP3' through the asset 'Production Planner' which is found in both the goal assignment of 'G2.2.1' and in the capacity of 'CAP3'.

4.3 Requirements modelling

The 4 different types of modelling shown in Fig. 5 represent the dimension of abstraction being applied. Orthogonal to this is the dimension of requirements lifecycle, which is a set of phases for progressing from an existing situation to a new desired situation, driven by the FCA's needs and wishes as well as perceived opportunities with respect to the CPPS technologies. The potential transformation is modelled in terms of change goals.

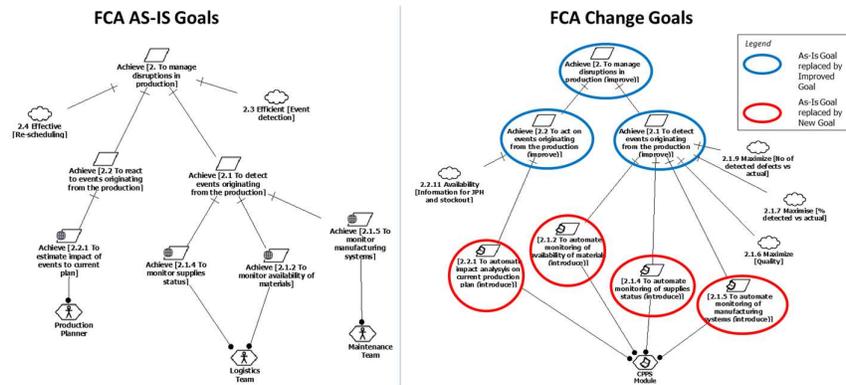


Fig. 6. Transformation from Current to Change FCA Goals

The change goals model is constructed in a top down stepwise manner, by generating the change goals either as *improvements* of the current goals or by *introducing* new goals. This process iterates on three main activities: (i) Determining the impact of perceived 'automation' on current business goals; (ii) Modifying the current goal hierarchy to reflect these changes; and (iii) Re-assigning operational goals to existing or foreseen actors (the CPPS modules). The result of this process is illustrated in Fig. 6, which depicts a segment of the change goal hierarchy corresponding to goal 'G2. To manage disruptions in production (improve)'.

As shown in this model the initial requirement to improve management of disruptions in production, is gradually operationalised through the introduction of a number of new goals, which are ultimately assigned to CPPS modules. These modules will represent a new set of assets and therefore capabilities. These new capabilities will give

FCA a competitive advantage, as well as dealing with current difficulties in solving problems mentioned in section 4.1 with their current set of capabilities.

Some of these system goals replace existing current goals, previously assigned to specific human actors. For example, change goal 'G2.1.5. To automate monitoring of manufacturing systems (introduce)' assigned to 'CPPS Module', replaces the current goal 'G2.1.5. To monitor manufacturing systems' currently assigned to the 'Maintenance Team'. Thus, it becomes obvious that the improvement sought by FCA will affect the dependencies between current actors and associated capabilities.

Therefore, the defined change goals express the CPPS requirements from a user perspective and motivate the desired transformation of FCA capabilities. For example, the FCA change goals (see Fig. 6) 'G2.1.5. To automate monitoring of manufacturing systems' and 'G2.2.1. To automate impact analysis on current production plan', motivate the introduction of a new capability 'CAP8: Smart Event Monitoring' and the transformation of an existing capability, that of 'CAP4: Production Scheduling', to that of 'CAP4'. This new situation is shown in the capability model of Fig. 7. The new capability 'CAP8: Smart Event Monitoring' comprises of three assets, namely the 'Cyber Physical System (CPS)', the 'Complex Event Processing (CEP)' and the 'Dashboard', with the abilities to 'monitor' the equipment condition, to 'diagnose' possible causes of failure and to 'inform' the shop floor manager.

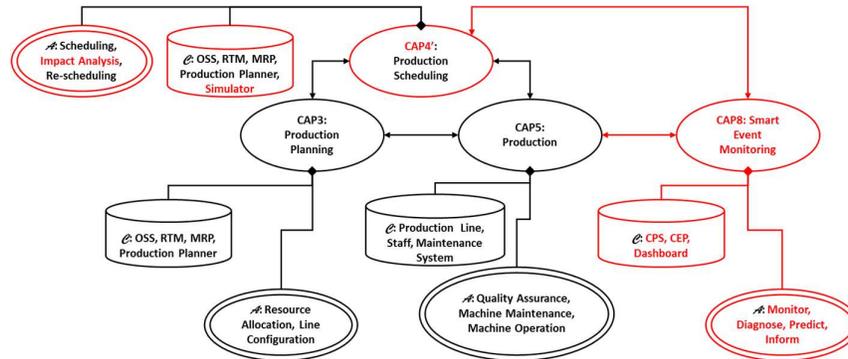


Fig. 7. Desired FCA Capabilities for the Production case

Similarly, the transformed capability 'CAP4' is augmented with the ability of 'impact analysis' that is the prediction of the effect of detected events on the production. This ability is effected through a new asset, that of the 'Simulator'. Note that these new assets, are elaborations of the CPPS modules mentioned above (see the Change Goal model in Fig. 6).

As shown in the model of Fig. 7, the new set of capabilities also gives rise to new collaborations between capabilities and involved actors. The identification of these relationships is significant because it enables us to define the way that actors coordinate between themselves in order to make capabilities realisable, further detailed in a new

actor model, as shown in Fig. 8. For example, the collaboration between the new capability 'CAP8' and the existing capability 'CAP5' gives rise to the 'Monitor Shopfloor' task dependency between the 'Shopfloor Manager' (a human actor) and the 'Dashboard' (a CPPS module) in the new actor dependency model.

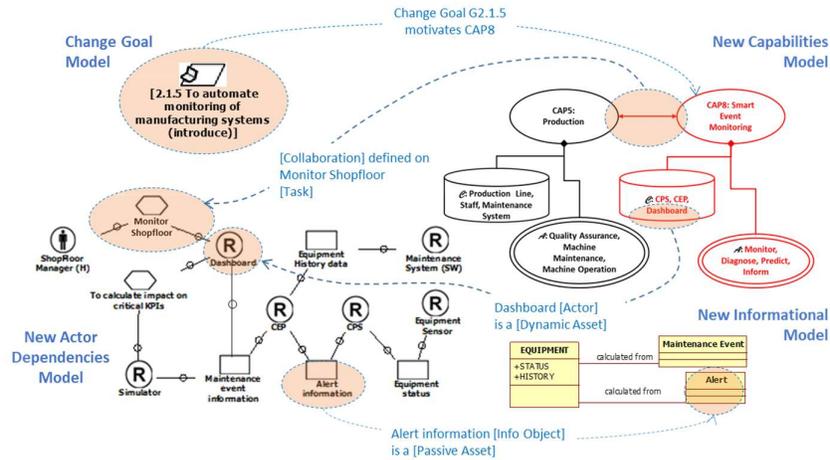


Fig. 8. Cross model relationships between change goals, desired capabilities and actors

Thus, starting from change requirements, new capabilities are modelled and these in turn give rise to details in a new actor model, which essentially defines operational requirements. This is even more explicit in the information model, also shown in Fig. 8, whereby the resource dependencies are elaborated. For example, the actor dependency on 'Alert information', which is a passive asset, leads to a revised information model that now incorporates the information object 'Alert'.

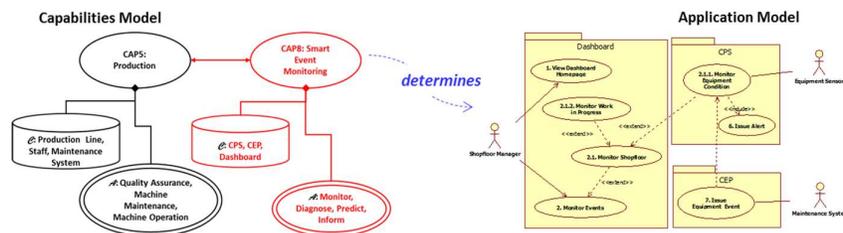


Fig. 9. Desired capabilities determine system models for late requirements

In summary, the process shown schematically in Fig. 8, provides a robust, analytical and traceable way of proceeding from high level strategic change goals to detailed operational requirements and the desired capabilities. Continuing with the capability driven paradigm, the identified desired capabilities determine the system components, in UML notation, as shown in Fig. 9.

5 Discussion

In this paper we have sought to present a systematic approach to RE for CPPS applications, an approach that has proved in practice to yield conceptual models that are (a) of value to end users, (b) consistent across all representations, (c) conducive to various analyses, (d) reflective of systemic impact of changes and (e) of value to developers of CPPS solutions. We have presented in this paper the conceptual foundations as well as the process phases of e-CORE and have demonstrated the approach's applicability on a specific industrial size application driven by the demand for dealing with both internal and external production disrupting factors. Through our capability-driven approach it is possible to capture the intertwined relationship among design requirements and environment, as well as the interplay of requirements and design artefacts [10].

Related research can be found in isolated sub-disciplines focusing for example on sensors, communications, networking, control theory, software engineering, computer engineering. In such approaches, systems are designed and analysed using a variety of modelling formalisms and tools whereby each representation highlights certain features and disregards others to make analysis tractable. These attempts may suffice to support a component-based approach to development but fails short on the intertwining of the multitude of agents whose interconnection is the very essence of cyber physicality. The importance of cooperating agents, has received attention in research work addressing intelligent systems [26]. However, this collaboration is considered at a physical level and these approaches are mostly concerned with disruption at the operational layer [27, 28]. Our approach complements the aforementioned works, by considering the influences of external factors on internal system networking and collaboration.

It is reported that work in cyber-physical systems pays less attention to early requirements [29, 30]. In our approach, through the identification of enterprise capabilities, it is possible to address fully early requirements, and to relate these seamlessly to late requirements. Furthermore, a key contribution of our approach is that a capability is considered in ontological terms and as such information about it is captured, represented and analysed in a methodical manner rather than either (a) regarding capability as some generic and nebulous construct or (b) considering the analysis in terms of other related constructs. This has the important implication that the RE process is indeed a capability-driven process.

Further to the early requirements issues discussed in this paper, we recognise that there are a number of significant CPPS challenges that still need to be addressed. These are linked to the specific nature of CPPSs which are neither static, nor isolated entities. CPPSs being composed of elements which may frequently constantly change impact their interconnectivity and their environment around them. Due to the scale of emerging CPPSs being Systems of Systems and the variety of events being generated, requirements cannot be expected to arrive in synchronised batches any longer, but rather as a consistent flow.

Part of CPPSs dynamicity is their mobility— their position and location are not fixed anymore, now they move in space and independently collaborate with other physical and cyber agents. Ocado and Amazon warehouses are great examples of the types of

industrial operations we may expect in coming years. Therefore, requirements engineering for CPPS should take mobility into account [31, 32].

CPPS dynamicity and mobility raise challenges with respect to quality properties of the live systems which need to be considered at requirements stage, properties that are often referred to as non-functional requirements (NFR) such as completeness, availability, reliability, consistency, relevance, and security [33]. We need to consider these NFRs as a way of monitoring and providing feedback on one hand from the system to the environment and on the other hand from the environment to the system.

Following the recommendations in [10], we recognise the following three cases of requirements challenges.

First, the *design* challenge related to the emergent behaviour and dynamics of the system and its environment. This implies that further to the fixed goals of the system which is the traditional way of specifying requirements we need to consider whether the system will continue to meet any emergent goals during the system's lifetime.

Second, the *modelling* challenge related to the anticipation and representation of the emergent behaviours of the system. In this sense we need to develop approaches for representing, communicating and analysing dynamic systems and their emergent requirements in ways that guarantee that these meet functional and non-functional requirements.

Third, the *predictability* challenge related to the impact that the system and its behaviour will have on its environment. In other words, we need to pay particular attention to the continuous dynamic composition of the system and its environment and how to predict the impact of the system on the environment, and vice versa.

The common thread in all these issues is that we need to define *self-awareness requirements* about the run-time success or failure or even the level and quality of service of other requirements. We posit that addressing such self-awareness requirements, is dealt by considering transformational capabilities that will lead to feedback functionality of the established system configuration.

To address these challenges we need models that should be capable of describing and reasoning about how the system monitors itself and its environment, how the system adapts itself and how the system coordinates monitoring and adaptation [34]. To this end, we envision extensions to the e-CORE approach by incorporating the notion of adaptation capabilities in its meta-model and its way-of-working. Such capabilities refer to run-time mechanisms that bring about changes to system capabilities defined at design time. Adaptation capabilities are triggered when the monitoring of NFRs (e.g. system availability, response time, etc.) detects signs of failures and to aid the system to reconfigure (predefined) system capabilities in order to restore the achievement of system goals. Adaptation capabilities may be considered as predictive or reactive, depending on the emergent properties that trigger the necessity for adaptation. In extending e-CORE we envision considering similar works in the area of requirements for self-awareness [35].

Acknowledgment

The notion of 'capability' in the context of digital enterprises was first investigated by the authors in the EU-FP7 funded project CaaS (# 611351). The e-CORE approach was developed as part of the Open Models Initiative and series of NEMO Summer Schools. It was extended and applied in the Qatar National Research Fund project i-Doha (# NPRP 7-662-2-247). Aspects of the FCA use case described in this paper were part of the work carried out by the authors for the EU H2020-FOF-11-2016 project DISRUPT (# 723541). The authors wish to express their gratitude to all their colleagues with whom they collaborated in the aforementioned projects.

References

1. Deloitte, *Industry 4.0: Challenges and solutions for the digital transformation and use of exponential technologies*. 2016.
2. IEC, *Factory of the Future - A White Paper*. 2016, IEC.
3. ITRE, *Industry 4.0: Study of the ITRE Committee*. 2016, Directorate General for Internal Policies, European Parliament.
4. Instrument Society of America, *Enterprise - Control System Integration Part 1: Models and Terminology*. 1999.
5. Instrument Society of America, *Enterprise - Control System Integration Part 3: Activity Models of Manufacturing Operations Management*. 2004.
6. Garetti, M. and L. Fumagalli. *P-PSO ontology for manufacturing systems*. in *14th IFAC Symposium on Information Control Problems in Manufacturing*. 2012. Bucharest, Romania.
7. Lemaignan, S., et al. *MASON: A Proposal For An Ontology of Manufacturing Domain*. in *IEEE Workshop on Distributed Intelligent Systems: Collective Intelligence and Its Applications (DIS'06)*. 2010.
8. Lin, H.K. and J.A. Harding, *A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration*. *Computers in Industry*, 2007. **58**: p. 428-437.
9. Fumagalli, L., et al. *Ontology-Based Modeling of Manufacturing and Logistics Systems for a New MES Architecture*. in *APMS 2014, Part I, IFIP AICT 438*. 2014.
10. Jarke, M., et al., *The Brave New World of Design Requirements*. *Information Systems*, 2011. **36**(7): p. 992-1008.
11. Loucopoulos, P., *Requirements Engineering for Emergent Application Software*, in *Lecture Notes in Business Information Processing (LNBIP)*, J. Cordeiro, L. Maciaszek, and J. Filipe, Editors. 2013, Springer. p. pp. 1-11.
12. Cao, J., et al. *Capability as Requirement Metaphor*. in *2011IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications*. 2011.
13. Danesh, M.H. and E. Yu. *Modeling Enterprise Capabilities with i*: Reasoning on Alternatives*. in *International Workshop on Advances in Services Design based on the Notion of Capability (ASDENCA'14)*. 2014. Thessaloniki, Greece: Springer.
14. Rosen, M. *Business Processes Start with Capabilities*. BP Trends 2010; Available from: <http://www.bptrends.com/publicationfiles/12-07-10-COL-BPM%20&%20SOA--BusProcesses%20begin%20with%20Capabilities%201003%20v01--Rosen.pdf>.
15. OMG. *Value Delivery Metamodel, v. 1.0*. 2013; Available from: <http://www.omg.org/spec/VDML/1.0/>.
16. DoD, *Systems Engineering Guide for Systems of Systems*. 2008: Washington DC.
17. MoD. *NATO Architecture Framework v4.0 Documentation*. 2013; Available from: <http://nafdocs.org/modem>.
18. Teece, D.J., *Dynamic Capabilities and Strategic Management*. 2009, United States, New York: Oxford University Press.
19. Helfat, C.E. and M.A. Peteraf, *The dynamic resource-based view: capability lifecycles*. *Strategic Management Journal*, 2003. **24**(997-1010).

20. Sandkuhl, K. and J. Stirna, eds. *Capability Management in Digital Enterprises*. 2018, Springer.
21. Barney, J., *Firm Resources and Sustained Competitive Advantage*. *Journal of Management*, 1991. **17**: p. 99-120.
22. Loucopoulos, P. and E. Kavakli, *Capability Oriented Enterprise Knowledge Modelling: The CODEK approach*, in *Domain-Specific Conceptual Modelling: Concepts, Methods and Tools*, D. Karagiannis, H.C. Mayr, and J. Mylopoulos, Editors. 2016. p. 197-215.
23. Loucopoulos, P. and E. Kavakli, *Capability Modeling with Application on Large-scale Sports Events*, in *AMCIS 2016*. 2016: San Diego, USA.
24. Loucopoulos, P. *Capability Modeling as a Strategic Analysis Tool - Keynote Extended Abstract*. in *IEEE Conference on Requirements Engineering: RePa Workshop*. 2016. Beijing, China: IEEE Computer Society.
25. Dimitrakopoulos, G., et al., *A Capability-Oriented Modelling and Simulation Approach for Autonomous Vehicle Management*. *Simulation Modelling Practice and Theory*, 2018.
26. Reza, H., et al. *Toward requirements engineering of cyber-physical systems: Modeling CubeSat*. in *2016 IEEE Aerospace Conference*. 2016.
27. Bertolino, A., et al., *A tour of secure software engineering solutions for connected vehicles*. *Software Quality Journal*, 2017.
28. Singh, I. and S.W. Lee. *Self-adaptive requirements for intelligent transportation system: A case study*. in *2017 International Conference on Information and Communication Technology Convergence (ICTC)*. 2017.
29. Braun, P., et al., *Guiding requirements engineering for software-intensive embedded systems in the automotive industry*. *Computer Science - Research and Development*, 2014. **29**(1): p. 21-43.
30. Chowdhury, N.M., L. Mackenzie, and C. Perkins. *Requirement analysis for building practical accident warning systems based on Vehicular Ad-Hoc Networks*. in *2014 11th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*. 2014.
31. Nelson, R., *Robots pick groceries, make sushi, assist amputees*. *EE-Evaluation Engineering*, 2018. **57**(2): p. 28-29.
32. Bogue, R., *Growth in e-commerce boosts innovation in the warehouse robot market*. *Industrial Robot: An International Journal*, 2016. **43**(6): p. 583-587.
33. Breivold, H.P. and K. Sandström, *Internet of things for industrial automation--challenges and technical solutions.*, in *2015 IEEE International Conference on Data Science and Data Intensive Systems 2015*, IEEE. p. 532-539.
34. Weyns, D., S. Malek, and J. Andersson, *FORMS: Unifying reference model for formal specification of distributed self-adaptive systems*. *ACM Transactions on Autonomous and Adaptive Systems (TAAS)*, 2012. **7**(2012): p. 1-61.
35. Souza, V., et al. *Requirements-Driven Qualitative Adaptation*. in *20th International Conference on Cooperative Information Systems (CoopOS'12)*. 2012. Rome, Italy: Springer-Verlag.