Thirteen concepts to play it safe with the cloud

Teun Hendriks
TNO-ESI,
Eindhoven, The Netherlands
Teun.Hendriks@TNO.nl

Benny Akesson
TNO-ESI,
University of Amsterdam,
Eindhoven, The Netherlands
Benny.Akesson@TNO.nl

Jeroen Voeten
Eindhoven University of Technology,
Eindhoven, The Netherlands
J.P.M.Voeten@tue.nl

Martijn Hendriks
Eindhoven University of Technology,
Eindhoven, The Netherlands
M.Hendriks@tue.nl

Javier Coronel Parada
Instituto Tecnólogico de Informática,
Valencia, Spain
jcoronel@iti.es

Miguel García-Gordillo
Instituto Tecnólogico de Informática,
Valencia, Spain
miguelgarcia@iti.es

Sergio Sáez
Instituto Tecnólogico de Informática,
Univ. Politécnica de Valencia
Valencia, Spain
ssaez@iti.es

Joan J. Valls
Instituto Tecnólogico de Informática,
Valencia, Spain
jvalls@iti.es

***Abstract*— Market trends show advanced usage of safety-critical systems with novel services based on smart data analytics. Customers require continuous updates to applications and services and seek lower costs, and easy-to-install solutions (maintenance) for safety-critical cyber-physical systems (CPS). Leveraging edge and cloud technologies has the potential to enhance safety-critical CPS, also in regulated environments. This is only possible when safety, performance, cybersecurity, and privacy of data are kept at the same level as in on-device only safety-critical CPS.**

This paper presents thirteen selected safety and performance concepts for distributed device-edge-cloud CPS solutions. This early result of the TRANSACT project aims to ensure needed end-to-end performance and safety levels from an end-user perspective, to extend edge and cloud benefits of more rapid innovation and inclusion of value-added services, also to safety-critical CPS.

Keywords—Safety-critical systems, cloud computing, edge computing, safety and performance concepts, performance management, service continuity, operational mode management

I. INTRODUCTION

Rapid innovation and incorporating AI-driven functionality are just two examples of how Cyber-Physical Systems (CPS) can benefit by being connected to the cloud. The TRANSACT project [1] investigates the transformation of safety-critical CPS from localized standalone systems into safe and secure distributed solutions leveraging edge and cloud computing towards such benefits. Nonetheless, this only flies when safety and performance of the system as a whole is ensured – as is privacy and security.

Consider a hospital, in which adding a cloud connection to medical imaging equipment, e.g., for minimally invasive treatment of patients, has the potential to support the task of a surgeon for treatment of patients. While safety-critical applications in such medical imaging (e.g., live X-ray imaging) will remain deployed in the device, mission-critical functionality, such as non-real-time image processing could be deployed advantageously in the cloud, as shown in Figure 1. During treatment, surgeons then may request the cloud to perform the latest innovations such as an 3D image processing analysis, to provide additional insight for the medical procedure at hand.

Considering safety and performance aspects is essential when such complex (cloud-based) image processing comes together with the need for application responsiveness during the live image guided treatment of patients. The cloud processing must not keep the surgeon (nor patient) waiting. Furthermore, in case of e.g., internet outages, the surgeon must always be able to revert to a well-defined degraded performance way of working.

This paper reports on an investigation [2] within the TRANSACT project as to which concepts are needed to ensure safety (mission-criticality) and performance an end-user perspective. With TRANSACT also concepts for the equally important security and privacy aspects were investigated [3], however, these are not in scope of this paper.

II. THE TRANSACT PROJECT

The overarching goal of the TRANSACT project [1] is to develop a universal, distributed solution architecture for the transformation of safety-critical CPS, from localized stand-alone systems into safe and secure distributed solutions leveraging edge and cloud computing. The TRANSACT project aims to leverage edge and cloud eco-systems to reduce CPS' cost and increase their pace of updates and improvements of applications and solution-oriented services.

The TRANSACT project aims to foster such creation of distributed solutions: incorporating device, edge, and cloud services within a TRANSACT reference architecture, and elaborated concepts and solutions for safety, performance, security, and privacy. These will be validated by means of demonstrators in five use cases from different application domains. Finally, a methodology for transition of stand-alone CPS to distributed solutions is envisaged as a key contribution of the TRANSACT project [1].

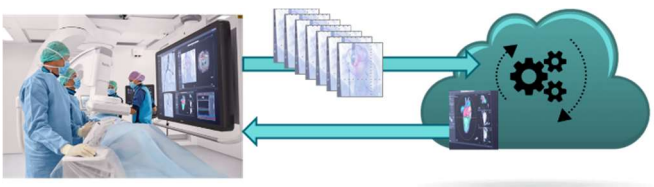

Figure 1. A cloud-connected advanced imaging workflow provides a surgeon access to the latest image processing capabilities.

This work was partially supported by the TRANSACT EU project, ((https://transact-ecsel.eu/). TRANSACT has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 101007260. The JU receives support from the European Union's Horizon 2020 research and innovation programme and Austria, Belgium, Denmark, Finland, Germany, Poland, Netherlands, Norway, and Spain.

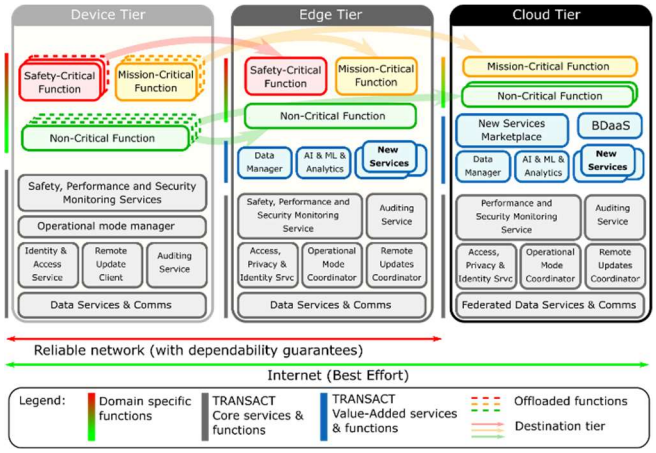

Figure 2.The TRANSACT reference architecture.

A. The TRANSACT reference architecture

The TRANSACT project has adopted a tree-tier, device-edge-cloud reference architecture [4]. This reference architecture defines the safety and mission critical functions, the core services and functions, and further value-added services and functions (see Figure 2). In brief, safety and mission critical functions are key in safety-critical CPS. Distributed safety-critical CPS solutions will be deployed over 3-tier (device-edge-cloud) architecture continuum. Each tier in the architecture provides a specific Quality-of-Service (QoS) level, especially with respect to performance aspects, such as response times and data-transfer guarantees, ranging from best effort to reliable and time-deterministic data transfers. Safety-critical functions often have hard real-time constraints, while mission-critical functions may have soft real-time constraints. In edge and cloud, value-added functions can be deployed including in the cloud Big Data as a Service (BDaaS) services.

The TRANSACT project innovates over stand-alone CPS by offloading functions to the edge or cloud tier with the purpose of improving quality and performance, and for improving innovation speed of the distributed CPS as new or upgraded functions can be deployed easier in edge and cloud.

However, when considering offloading functions from the device it is critical to ensure a CPS system's end-to-end safety, performance, security, and privacy. Therefore, a number of dedicated core services are introduced to cooperatively realize that objective. The safety, performance, and security monitoring services handle detection, identification, and prevention of safety, security, and performance failures. In addition, they track the Service Level Indicators (e.g., latency, throughput, availability) that are used by the operational mode manager (running on the device) and the operational mode coordinator (running at the edge/cloud tier) to decide at runtime whether a device's function can be executed remotely or not. Other core services address additional security and privacy concerns and take care of updates.

B. TRANSACT use cases for distributed CPS solutions

In the TRANSACT project, five use cases will experiment with the reference architecture, its components, and a mix of the selected safety, performance, security, and privacy concepts with the aim to capture the overarching results across the various use cases. This allows the TRANSACT project to validate the approach and refine the proposed reference architecture over the course of the project.

These five Use Cases (UC) cover multiple application areas [1]. The first UC concerns fleets of remote controlled, (semi-)autonomous vehicles in urban areas. The envisaged solution will allow vehicles to be moved from one location to another without a driver, but with assistance of a remote operator. The operator will receive continuous feedback on vehicle state and environment, allowing him/her to assist the vehicle to navigate through urban traffic.

The second UC concerns critical maritime decision support enhanced by distributed, AI-enhanced edge and cloud solutions. Today's vessels are mainly "safety islands" with only limited support from outside. The envisioned cloud (shore) - device (vessel) solution offers AI-enhanced, near real-time (shore-based) monitoring of the vessels related to safe and efficient navigation, and warning and decision support issued to vessels.

The third UC concerns the safety of the increasingly electrified European car fleet. Cloud-based battery management offers fleet-based analysis of usage for the improvement of functionality (e.g., time left to charge), safety or autonomous driving (e.g., fail-operation in the battery/battery management system).

The fourth UC concerns cloud-connected image guided therapy (see also Figure 1). Offloading non-safety critical functionality to the edge-cloud platform solution is a key enabler for new business models as additional services (like AI-based solutions) and allows usage of the latest versions of clinical applications in the installed base of equipment.

The fifth UC concerns connected wastewater treatment plants. These are key to mitigate climate change induced water scarcity while preventing ecological disasters due to potential wastewater spills. Cloud-based data aggregation from different facilities in a single management tool allows obtaining comparative indicators between different facilities and improves failure detection, increases the understanding of water processes and, finally, leads to better water quality.

For all these use cases, transforming the safety critical local CPS into distributed solutions based on the functionality (applications and services) deployed over the device-edge-cloud continuum (see Figure 2) is crucial. The variety of application domains helps the TRANSACT project to deliver generically applicable results.

III. THIRTEEN CONCEPTS TO PLAY IT SAFE WITH THE CLOUD

Distribution of mission-critical functionality, and in the first use case, safety-critical functionality, to edge and cloud, is key for the envisaged solutions and innovations in the use cases. However, when migrating functions from device to edge or cloud the end-to-end safety, performance, security, and privacy of the distributed safety-critical CPS solution must be preserved.

In the first year, the TRANSACT project has performed a thorough selection of concepts for safety, performance, security, and privacy. We present here the result of the investigation into the end-to-end safety and performance concepts for distributed safety-critical CPS solutions. The evaluation has reflected relevant safety and performance concerns, the TRANSACT reference architecture, and needs stemming from the use cases. The result is a public TRANSACT deliverable [2], reporting on necessary concepts for managing performance and safety for deploying and

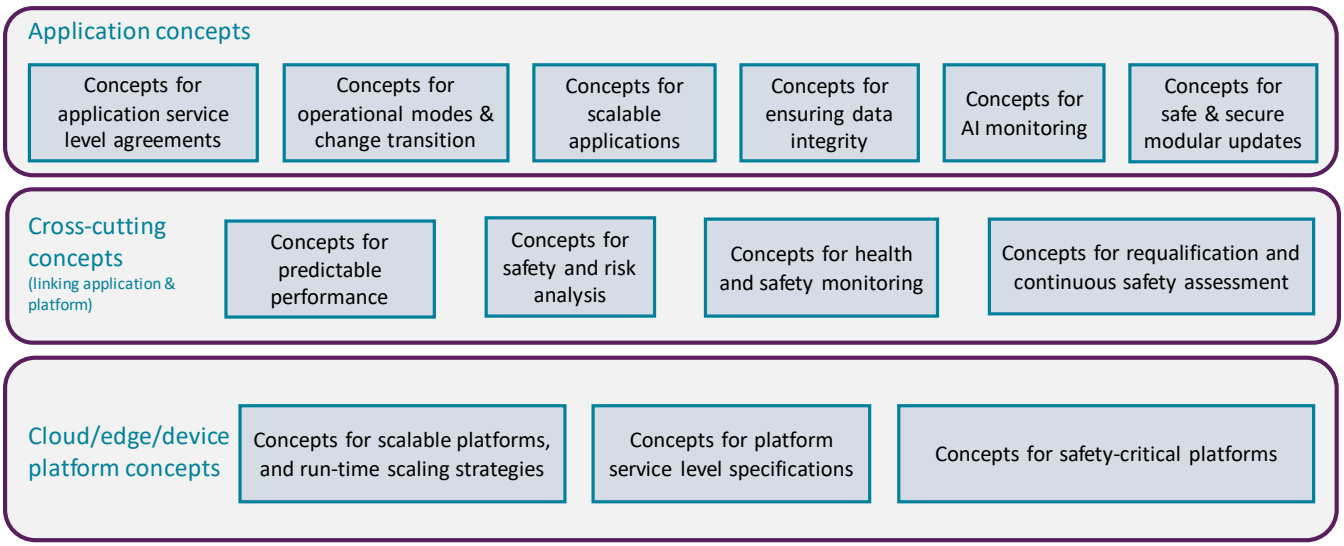

Figure 3. Overview of thirteen identified safety and performance concepts types to consider when connecting CPS to the cloud.

running distributed applications for safety-critical device-edge-cloud type systems.

A. *Three categories to cluster selected concepts*

The selected concepts are clustered into three main categories: application concepts, cross-cutting concepts, and platform concepts. These are defined as follows:

- **Application concepts** target the Application: the functionality that implements a particular solution to help the end-user perform a specific task. An application can be composed of a monolithic service or a group of distributed services that are executed in different and distributed targets in the device, edge, cloud continuum.
- **Platform concepts** target the Platform: the environment in which the application is executed. It comprises of the complete infrastructure in the device, edge, cloud continuum to execute an application, including hardware, operating systems, hypervisors, communication networks, containers, and cloud computing services.
- **Cross-cutting concepts** are system-level methods and techniques for linking application and platform. They include concepts for designing and deploying the application on the platform, or for analyzing, monitoring and managing the distributed CPS solution's run-time behavior.

This clustering follows the so-called Y-chart approach [5] [6], which models application functionality and the implementation platform as separate elements, with an explicit mapping as a variation point — to be addressed by the cross-cutting concepts. This categorization fosters the separation of concerns and eases design-space exploration.

B. *Selected concepts for performance and safety*

For each of the above categories, a number of concept classes have been identified (see Figure 3). Within each concept class, selected concepts and methods are described [2] and further research challenges identified. Here we present a summary of the concepts and related challenges.

On **application level**, the TRANSACT project has identified the following concepts:

1) Concepts for application service level agreements. Service level expectations need to be specified and managed *[7]*. In particular when deploying AI services in the cloud, concepts need to be developed to fit with the use of AI in distributed safety-critical CPS solutions. A special challenge here is to find suitable metrics that can measure AI model prediction accuracy versus the need of retraining in context of safety-critical usage scenarios.

2) Concepts for operational modes and change transition. As edge and cloud, in general, have no absolute availability guarantees, on-device fallback functionality and seamless change-over to fallback functionality need to be managed, so that safety is always warranted. This requires applications to offer various operating modes *[8]*, and support seamless change btween operation modes. A special challenge in a distributed multi-mode solution is to ensure consistency over the continuum at each different operational mode, and synchronisation of each possible transition between modes over the continuum.

3) Concepts for scalable applications. Scalability of applications is one of the key enablers for operational mode management and change transition. A concept is identified to achieve this by trading-off an application's QoS level with the resources available to the application. A special challenge here is to ensure predictability of resource (re-) allocation across the device-edge-cloud continuum (see also *[9]*).

4) Concepts for ensuring data integrity. In distributed safety-critical CPS solutions, safety and privacy related data leaves the confines of the device, and vice versa external data is imported for use in safety-critical or mission-critical functions. This requires the data integrity to be established and safeguarded. A special challenge here is to ensure adequate data integrity monitoring even with intermittent and low-bandwidth cloud connectivity.

5) Concepts for AI monitoring. The cloud is an excellent place to deploy advanced AI services. Yet, post-deployment, AI model performance can degrade over time for several reasons including data drift *[10]*. Concepts are identified to monitor AI performance in service, to ensure fit for use, or to initiate retraining. A special challenge here is how to combine the need for extensive data for AI monitoring

with security, privacy and trust in highly regulated environments.

6) Concepts for safe and secure modular updates. To reap the benefits of the cloud, it must be possible to update functionality across the device-edge-cloud continuum incrementally in a safe, secure, and modular manner. A special challenge here is how to assure in-the-field correctness of AI-based services with a combination of test-time and run-time verification and re-qualification techniques.

To address **cross-cutting concerns**, i.e., to ensure that applications use the platform properly, the following concepts have been identified:

7) Concepts for predictable end-to-end performance. Ensuring end-to-end performance across the device-edge-cloud continuum requires performance analysis, monitoring, prediction and management *[11]*, and this across a distributed device-edge-cloud solution. A special challenge here is how to cope with the more stochastic nature of cloud performance, to obtain adequate models and measurements needed as input end-to-end for performance management.

8) Concepts for safety and risk analysis. Traditional methods such FMEA typically consider the impact of (individual) component failures. However, in distributed solutions, also undesired interactions between otherwise fine components could cause safety issues. A special challenge here is how to analyse emergent risks based on undesired device-edge-cloud interactions. Concepts such as STPA *[12]* can be tailored *[13] [14]* to analyse safety and security risks.

9) Concepts for health and safety monitoring. When safety-critical or mission-critical functions are offloaded to the edge or cloud, then *distributed* monitoring at run-time is necessary to ensure that the total system is operating safely. Such monitoring can encompass the availability of the device, edge, cloud, as monitoring liveness, responsiveness and integrity of applications. A special challenge here is how to achieve integral monitoring across device, edge, and cloud.

10) Concepts for requalification and continuous safety assessment. A key aspect for achieving rapid innovation is that the necessary proof for safety assurance of incremental updates can be provided efficiently, i.e. to be suitable for low-effort, reduced cost, and frequent re-qualification. A special challenge here is how to achieve modular safety evidence, that can be integrated in a DevOps pipeline *[15]* to foster incremental innovation.

Finally, **on platform level**, a distributed cloud-edge-device platform needs to offer the necessary predictable and scalable configuration for the distributed system to be performant and to function safely. The following concepts have been identified:

11) Concepts for scalable platforms and run-time scaling strategies. Cloud platforms provide an efficient infrastructure to build high-performing and scalable distributed applications, and provide scaling options to accommodate high or low demanding workflows. Concepts are identified to manage the dynamic demand of a varying number of devices by a (scalable) cloud platform. A special challenge here is how to transform the monolithic architecture of stand-alone CPS to fit with the service-oriented cloud architectures *[16]*.

12) Concept for platform service level specifications. Concepts are identified to ensure proper usage of a well configured platform's resources and assets such that a distributed device edge-cloud solution fulfils safety, performance, security, and privacy requirements. A special challenge here is how to deploy applications across regions and availability zones with dependable communication to minimise the impact of failures on critical system functionality and achieve high resiliency.

13) Concepts for safety-critical platforms. Safety-critical or mission-critical functionality requires special means and measures [17] [18] to achieve fail-safe or fail-operational behavior. When such functionality is offloaded to the edge or cloud, then these parts of the distributed solution (and their interlinked communication and network) must employ measures to support such safety-critical or mission-critical functionality. A special challenge here is how to distribute the safety-relevant (mission-relevant) measures, redundancy, and monitoring over the device-edge-cloud continuum.

The TRANSACT deliverable [2] describes more fully each concept and how it fits in the TRANSACT reference architecture, with an example of its application in one of the TRANSACT use cases. Furthermore, challenges for application of the concept in the device-edge-cloud continuum type of systems are listed: these form the basis of further investigation in the scope of the TRANSACT project.

IV. Integrating Concepts into a Solution

TRANSACT UC 4, cloud-connected Image-guided Therapy, presents a good example of how these concepts need to come together to realize a good, distributed safety-critical CPS solution as a device-cloud continuum.

Figure 4 shows the workflow of the cloud-assisted advanced image analysis with many hospitals concurrently. During operations surgeons may want to request one or more advanced image analysis tasks to be processed in the cloud, say every few minutes. These cloud requests are dynamically executed on cloud resources, e.g., as part of an auto scaling group. With many thousands of requests for GPU-intensive image processing in the cloud, the cloud side needs to be dynamically scaled to handle variations and peaks in load. This load is not fully predictably nor completely stochastic, as operations typically start at the top of the hour and proceed according to an operation protocol. Variations exist though. Auto scaling groups can be set up for this purpose but need to be managed well. Yet, in case of cloud disruptions, surgeons need to be able to proceed with the treatment.

In the following, we illustrate how several concepts need to come together to achieve predictable performance, maintain service continuity, and manage operational modes, all to create a performant and safe distributed solution. We highlight aspects of current research ongoing in the TRANSACT project.

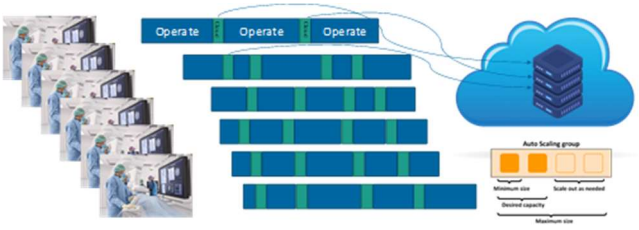

Figure 4. Concurrent hospital usage of cloud-connected Imaged-Guided Therapy.

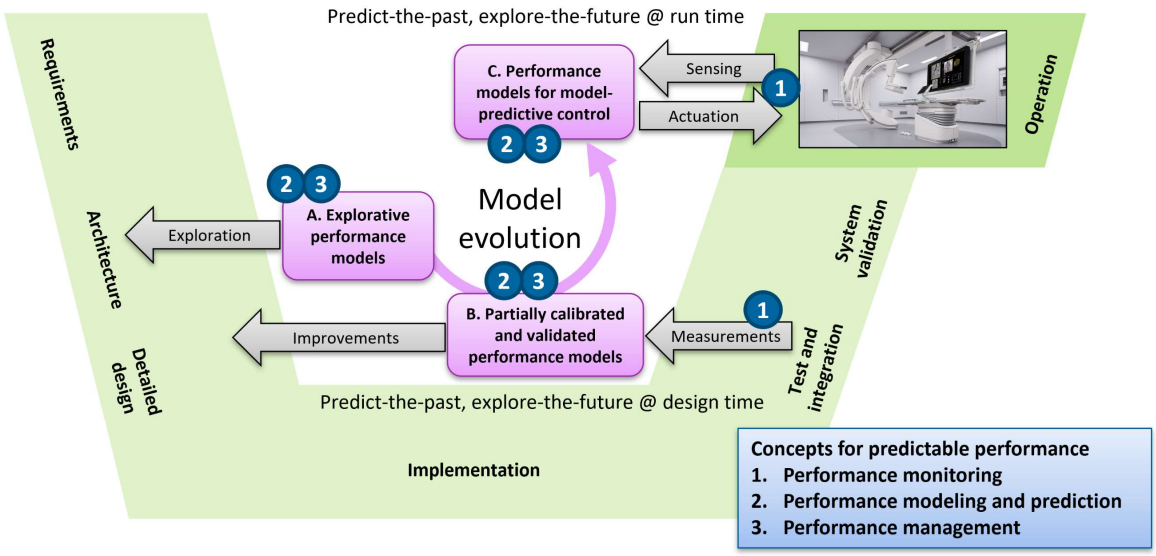

Figure 5. Concepts for predictable performance in the system lifecycle.

A. *Achieving predictable performance*

In the concepts for predictable performance, three detailed concepts have been identified based on [11]: *performance modeling and prediction; performance monitoring,* and *performance management.* These concepts contribute to model-driven system-performance engineering in which performance models play a pivotal role: they are a single source of truth and form a basis for the automated synthesis of implementation artifacts. In the TRANSACT project, we are exploring these three concepts in the context of various phases of the system lifecycle (simplified with the V model), see Figure 5.

The concept of *performance modeling and prediction* is central. It can be used during design-time and run-time. Explorative performance models can be made early during system development; these can be used to explore qualitative system behavior. When data becomes available from *performance monitoring* of implemented components during test and integration, the performance models can be refined, calibrated and validated. Insights from model analysis can be reason for further system improvements. Finally, the calibrated performance models can be used in combination with the monitoring for run-time *performance management*.

In the TRANSACT project, we have made several of these elements concrete. First, we have created a high-level discrete-event performance model of UC 4 with the POOSL toolset [19] for the purpose of architectural exploration. The model follows the Y-chart pattern [6] and includes a piecewise-linear progress abstraction for shared bandwidth [5]. Furthermore, a stochastic workload model is used, which results in a burst load for the communication and computing platform. Faithful modelling of distributed safety-critical cyber-physical systems on a high level of abstraction with POOSL seems feasible, modelling patterns from other domains can be reused, and we see more cloud-specific modelling patterns emerge. Furthermore, response-time properties can be formally expressed and analysed. Within the TRANSACT project, we want to further explore how model calibration can be systematized (e.g., in an online setting), and how performance models can be used for runtime performance management.

Performance management across the continuum is closely aligned with operational mode management (see also section C, Operational mode management). For modeling key qualities and resources in UC4, we used the Quality and Resource Management Language (QRML, [20] [21]), in both the edge and the cloud tier. A question that we want to explore in the TRANSACT project is how to integrate performance models that have different models-of-computation with QRML, to facilitate system-level multi-objective optimization, performance management, in relation to operational mode management across the continuum.

B. *Maintaining service continuity*

Maintaining service continuity is essential when complex (cloud-based) image processing comes together with the need for application responsiveness during the live image guided treatment of patients. During a temporary cloud service disruption or spurious failure [22], a surgeon must always to be able to revert to a well-defined, and if necessary, device-only, way of working.

For mission-critical applications, a device's response to cloud service disruptions must be considered and triggered appropriately. Ensuring service continuity while offloading mission-critical functionality to the cloud requires a *safety-critical platform concept* to be integrated with *health and safety monitoring concepts* (in combination with performance and security monitoring services) as *operational mode management concepts,* in co-operation with *concepts and solutions for predictable performance.*

Figure 6 shows how a combination of concepts needs to come together to achieve service continuity and predictable performance in UC 4 (instantiating the TRANSACT reference architecture for a device-cloud continuum – see Figure 2).

Safety-critical functionality remains positioned on the device. Yet for mission-critical functionality, the primary actuation channel is positioned in the cloud to perform the required functionality. The fail-safe mission-critical processing channel is positioned in the device. This channel performs needed fallback functionality in case the primary channel is not available or otherwise not according operating in line with the mission policies.

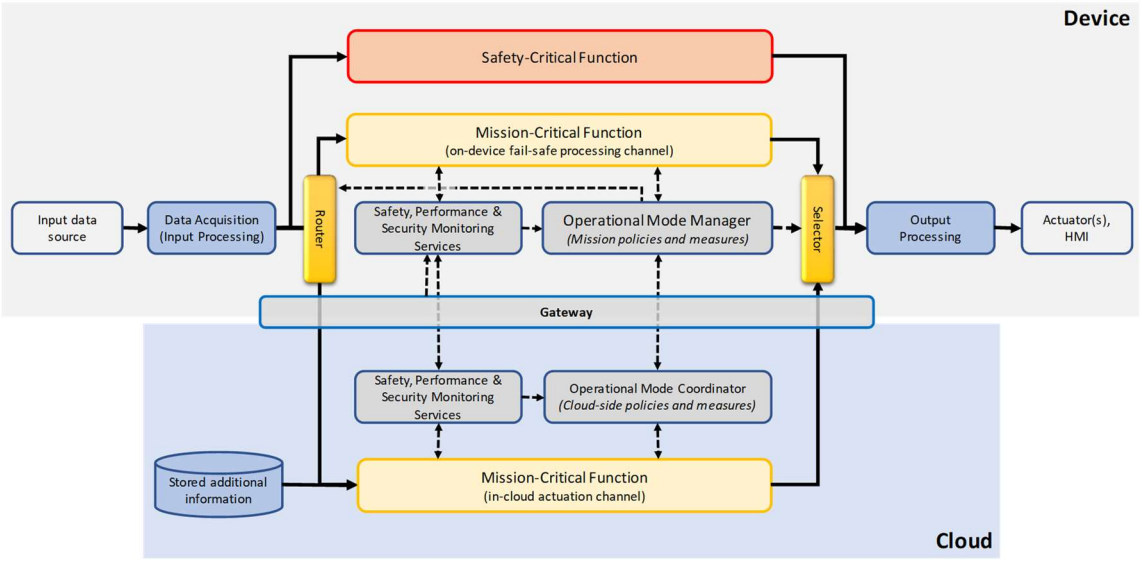

Figure 6. Service continuity in a device-cloud continuum requires a performant, yet fail-operational solution for mission-critical functionality utilising distributed monitoring and operational mode management.

On-device side, the Operational Mode Manager takes on the central role of a safety executive [17] [18] coordinating all mission-measures required to switch over to the fail-safe processing channel or bring the system to a safe state. The health and safety monitoring services check the liveness and proper functioning of the actuation channel and, when compromised, signal an anomaly to the Operational Mode Manager. The Operational Mode Manager then manages the routing, mode change, and the transition of control (see section C, Operational mode management), finally selecting which of the mission-critical channels to use towards the actuators and/or HMI.

The distributed solution elaboration is now investigated in the TRANSACT project, drawing on prior distributed architectures such as in fail-operational Truck Platooning [23]. Two key challenges are the following:

- How to distribute the mission-relevant operation, redundancy, and monitoring over the device-edge-cloud continuum?
- How to keep service continuity solutions stable over edge or cloud-side upgrades meant to support faster innovation and independent releasing of new/upgraded edge or cloud functionalities?

In the TRANSACT project, the use of STPA [12] is investigated to guide the design of device-edge-cloud continuum's control and monitoring structures on basis of a conceptual architecture [24] capturing the fail-operational safety executive concept [17] [18]. The need for modular V&V in such solutions us also investigated such that cloud-side innovations can be integrated in a DevOps pipeline [15].

C. *Operational mode management*

For distributed device-edge-cloud solutions, switching from the current operational mode to another one (e.g., migrating mission-critical functionality from cloud to the device to maintain service continuity) requires substituting the current executing tasks with those of the target mode. This process could introduce a transient unstable stage, where tasks of both old and new modes may coexist. This could lead to an overload which may compromise the system schedulability (i.e., the ability to meet all timing constraints of the system). This is highly undesirable in real-time systems, where the correctness of the system not only depends on producing correct output but also on providing it at the right time [17]. Schedulability must be ensured at each different operational mode and in each possible transition between modes. This is even more complex in distributed CPS solutions, where some components may, at some point, still be unaware of a mode transition being in place due to messages not having arrived yet. Coordination between the components is required to safely perform a mode change.

Both the Operational Mode Manager, at the device tier, and the Operational Mode Coordinator, at the edge-cloud tier, in the TRANSACT reference architecture are responsible of guaranteeing a safe transition during a mode change. This transition is initiated by a Mode Change Request (MCR) and needs to be managed following a Mode Change Protocol (MCP), which defines the allowed transitions, the type of transitions, the offset times and other conditions and constraints to guarantee the timing requirements of the system.

In the example of the previously described UC 4, if the anticipated response time of the cloud-based 3D image processing analysis cannot be met for some reason (e.g.,

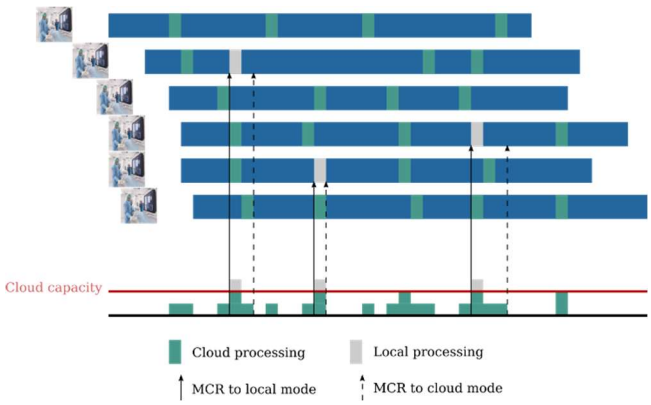

Figure 7. Concurrent hospital with different devices running on cloud mode or local mode, depending on the cloud occupancy.

network unavailability or limited bandwidth), the system should recognize this situation and switch to another operational mode to mitigate the problem. This directly relates to the *concepts for health and safety monitoring* and to the *concepts for predictable performance*, and will decide the moment in time when an MCR must be issued. Figure 7 depicts an example of a multi-mode application for this specific scenario, where images are processed either in the cloud or locally in the device, depending on the cloud availability. The device itself will change to local mode if communications with the cloud do not comply with QoS, otherwise, the Operational Mode Coordinator, executed in the cloud, will send a request to any of the devices to change to a device-local mode of operation to prevent the capacity of the cloud from being surpassed.

Research for solutions for the Operational Mode Manager and Operational Mode Coordinator in the TRANSACT architecture has begun. Regarding the first, we propose a generic multi-platform middleware, that can be reused in several systems that are built using this architecture. It is based on the directives described in SafeMC [25] that allow the service to configure multiple MCPs. Similarly, a generic solution, designed as a middleware library, is being developed to handle the synchronization of multi-mode applications as part of the latter.

V. CONCLUSIONS

When connecting safety-critical systems to the edge and cloud, it is of paramount importance to ensure end-to-end safety, performance, security, and privacy. This paper has reported on the selection and evaluation of relevant end-to-end safety and performance concepts for distributed safety-critical CPS solutions in the context of the TRANSACT project.

This paper has presented thirteen selected concepts in support of ensuring end-to-end safety and performance in distributed device-edge-cloud CPS solutions, with research challenges listed. In the TRANSACT project, these concepts are currently being elaborated into solutions, which may be more specific to various domains and realizations of device-edge-cloud continuum systems. To illustrate how these concepts need to come together in distributed safety-critical CPS solutions, we provided an example from cloud-connected image guided-therapy and highlighted some of the ongoing research in the TRANSACT project.

ACKNOWLEDGMENT

The authors would like to thank all TRANSACT project partners for their contributions to the TRANSACT reference architecture; the presented concepts; and the fruitful collaboration and the ongoing elaboration of those concepts in the TRANSACT use cases and demonstrators.

REFERENCES

[1] TRANSACT EU project, "The TRANSACT project," 2022. [Online]. Available: https://transact-ecsel.eu/.

[2] TRANSACT EU project, "D8 (D3.1) Selection of concepts for end-to-end safety and performance for distributed CPS solutions," TNO, 30 05 2022. [Online]. Available: https://transact-ecsel.eu/resources.

[3] TRANSACT EU project, "D9 (D3.2) Selection of concepts for end-to-end security and privacy for distributed CPS solutions," DTU, 30 05 2022. [Online]. Available: https://transact-ecsel.eu/resources.

[4] TRANSACT EU project, "D7 (D2.1) Reference architectures for distributed safety-critical distributed cyber-physical systems v1," ITI, 30 05 2022. [Online]. Available: https://transact-ecsel.eu/resources/.

[5] M. Hendriks, T. Basten, J. Verriet, M. Brassé and L. Somers, "A Blueprint for System-Level Performance Modeling of Software-Intensive Embedded Systems," *Software Tools for Technology Transfer 18(1),* pp. 21-40, 2016.

[6] F. Balarin, M. Chiodo, P. Giusto, H. Hsieh, A. Jurecska, L. Lavagno, C. Passerone, A. Sangiovanni-Vincentelli, E. Sentovich, K. Suzuki and B. Tabbara, Hardware-Software Co-design of Embedded Systems: The POLIS Approach, Springer New York, 1997.

[7] B. Beyer, C. Jones, J. Petoff and N. R. Murphy, Site Reliability Engineering: How Google runs production systems, O'Reilly Media, Inc., 2016.

[8] A. Burns, "System mode changes-general and criticality-based," in *2nd Workshop on Mixed Criticality Systems*, 2014.

[9] S. Ferretti, V. Ghini, F. Panzieri, M. Pellegrini and E. Turrini, "Qos–aware clouds," in *IEEE 3rd International Conference on Cloud Computing*, 2010.

[10] M. Arnold, J. Boston, M. Desmond, E. Duesterwald, B. Elder, A. Murthi, J. Navratil and D. Reimer, "Towards automating the AI operations lifecycle," *arXiv preprint,* vol. arXiv:2003.12808, 2020.

[11] v. d. B. Sanden, Y. Hi, v. d. J. Aker, B. Akesson, T. Bijlsma, M. Hendriks, K. Triantafyllidis, J. Verriet, J. Voeten and T. Basten, "Model-driven system-performance engineering for cyber-physical systems," in *International Conference on Embedded Software (EMSOFT)*, 2021.

[12] N. G. Leveson, Engineering a safer world: Systems thinking applied to safety, Boston: The MIT Press, 2016.

[13] I. Friedberg, K. McLaughlin, P. Smith, D. Laverty and S. Sezer, "STPA-SafeSec: Safety and security analysis for cyber-physical systems," *Journal of information security and applications,* vol. 34, pp. 183-196, 2017.

[14] J. Yu, S. Wagner and F. Luo, "Data-flow-based adaption of the System-Theoretic Process Analysis for Security," *PeerJ Computer Science,* vol. 7, no. e362, 2021.

[15] M. Zeller, "Towards Continuous Safety Assessment in Context of DevOps," *arXiv preprint arXiv:2106.07200,* 2021.

[16] S. Newman, Monolith to Microservices, Sebastopol, CA: O'Reilly, 2019.

[17] B. P. Douglass, Real-time design patterns: robust scalable architecture for real-time systems, Addison-Wesley Professional, 2005.

[18] A. Armoush, *Design Patterns for safety-critical embedded systems,* Aachen: RWTH Aachen University, 2010.

[19] "POOSL," 2022. [Online]. Available: https://www.poosl.org.

[20] v. d. F. Berg, V. Čamra, M. Hendriks, M. Geilen, P. Hnetynka, F. Manteca, P. Sánchez, T. Bureš and T. Basten, "QRML: A Component Language and Toolset for Quality and Resource Management," in *Forum on specification & Design Languages, FDL 2020, Proceedings*, Kiel, Germany, 2020.

[21] M. Hendriks, M. Geilen, K. Goossens, R. de Jong and T. Basten, "Interface Modeling for Quality and Resource Management," *Logical Methods in Computer Science, LMCS,* no. 19, pp. 1-34, 26 May 2021.

[22] P. Huang, C. Guo, L. Zhou, J. R. Lorch, Y. Dang, M. Chintalapati and R. Yao, "Gray failure: The achilles' heel of cloud-scale systems," in *Proceedings of the 16th Workshop on Hot Topics in Operating Systems*, Whistler, BC, 2017.

[23] T. Bijlsma and T. Hendriks, "A fail-operational truck platooning architecture," in *IEEE Intelligent Vehicles Symposium (IV)*, Redondo Beach, CA, 2017.

[24] N. Levenson, "An Improved Design Process for Complex, Control-Based Systems Using STPA and a Conceptual Architecture," Massachusetts Institute of Technology, Boston, MA, 2019.

[25] T. Chen and L. T. X. Phan, "SafeMC: A system for the design and evaluation of mode-change protocols," in *2018 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, 2018.