

# A Cloud-based Collaboration Platform for Model-based Design of Cyber-Physical Systems

Peter Gorm Larsen<sup>1</sup><sup>a</sup>, Hugo Daniel Macedo<sup>1</sup><sup>b</sup>, John Fitzgerald<sup>2</sup><sup>c</sup>, Holger Pfeifer<sup>3</sup>,  
Martin Benedikt<sup>4</sup>, Stefano Tonetta<sup>5</sup><sup>d</sup>, Angelo Marguglio<sup>6</sup>, Sergio Gusmeroli<sup>7</sup> and George Suciu Jr.<sup>8</sup>

<sup>1</sup>*DIGIT, Department of Engineering, Aarhus University, Aarhus, Denmark*

<sup>2</sup>*School of Computing, Newcastle University, U.K.*

<sup>3</sup>*fortiss, Germany*

<sup>4</sup>*Virtual Vehicle, Austria*

<sup>5</sup>*Fondazione Bruno Kessler, Italy*

<sup>6</sup>*Engineering Ingegneria Informatica S.p.A., Italy*

<sup>7</sup>*Politecnico di Milano, Italy*

<sup>8</sup>*BEIA Consult, Romania*

**Keywords:** Model-based Design, Tools, Models, Collaboration Platform.

**Abstract:** Businesses, particularly small and medium-sized enterprises, aiming to start up in Model-Based Design (MBD) face difficult choices from a wide range of methods, notations and tools before making the significant investments in planning, procurement and training necessary to deploy new approaches successfully. In the development of Cyber-Physical Systems this is exacerbated by the diversity of formalisms covering computation, physical and human processes. In this paper, we propose the use of a cloud-enabled and open collaboration platform that allows businesses to offer models, tools and other assets, and permits others to access these on a pay-per-use basis as a means of lowering barriers to the adoption of MBD technology, and to promote experimentation in a sandbox environment.

## 1 INTRODUCTION

The digital transformation that industry and society is experiencing creates many opportunities for increasing the delegation of tasks to machines. The dependability of the resulting systems therefore becomes critical not only for compliance with regulations and certification standards, but also for societal acceptance of the transformation. Modern innovative products and systems combine physical and networked computational processes. Such Cyber-Physical Systems (CPSs) place new multi-disciplinary demands on traditional engineering processes because of the heterogeneity and complexity of the constituent elements, and of their interaction with the environment.

Model-Based Design (MBD) has demonstrated the potential to increase the quality of CPSs (Van der Auweraer et al., 2013). MBD prescribes the use of system models through the development process in order to represent system structure and behaviours, providing a basis for machine-assisted analysis of system properties, and informing design decisions through processes of refinement into implementation.

A considerable body of research has provided model-based solutions to the challenges of CPS design (Beckers et al., 2007; Sztipanovits et al., 2015), but businesses that could benefit from such approaches may face barriers to their adoption. It is possible that as a consequence, MBD methods and tools appear largely to be applied in domains such as aerospace where the return of investment can take decades. By contrast, Small and Medium-sized Enterprises (SMEs) require considerable flexibility to change processes to adopt MBD, and may lack in-house expertise. In addition, the selection, procure-

<sup>a</sup> <https://orcid.org/0000-0002-4589-1500>

<sup>b</sup> <https://orcid.org/0000-0002-8882-4500>

<sup>c</sup> <https://orcid.org/0000-0001-7041-1807>

<sup>d</sup> <https://orcid.org/0000-0001-9091-7899>

ment, training and deployment costs for some methods and tools can be discouragingly high. In general, it is difficult for SMEs to invest in acquiring the necessary background for example because of the high license fees from commercial vendors of MBD assets.

In this position paper, we report on an approach that aims to make MBD more accessible to a range of businesses, but especially SMEs, involved in the development of cyber-physical products and systems. This centres on the servitisation of modelling, simulation and analysis tools for CPS design, offered through an open collaboration platform. The new HUBCAP project<sup>1</sup> aims to implement and evaluate this approach, alongside provision of other services for SMEs through a network of innovation hubs. In this paper we focus on the platform.

The goal of our HUBCAP project is to provide a collaboration platform that enables users to access advanced CPS design and engineering solutions, including models and tools such as those offered, for example, by the Modelica association community (Fritzon, 2015) or the INTO-CPS association community (Larsen et al., 2016). This will be done through *sandboxes* – environments in which users can test and experiment with a solution in a secure and trusted environment before investing in longer-term or larger-scale adoption. The platform will also facilitate collaboration services that enable the sharing of knowledge among providers of MBD solutions, so that new models, tools and techniques and related services may be extended by combining existing assets.

This paper provides an overview of our proposed collaboration platform in Section 2, and more detail on the sandbox functionality in Section 3. Section 4 indicates how assets such as models can be accessed from a catalogue inside the collaboration platform. Afterwards Section 5 explains how we envisage the collaboration platform to be populated with MBD assets by means of open calls targeted at SMEs, enabling them to get financial support. Finally, we conclude in Section 6 by discussing the work required to realise and evaluate such a platform.

## 2 THE COLLABORATION PLATFORM

The HUBCAP Collaboration Platform is based on the DIHIWARE open source solution developed by the MIDIH H2020 EU project<sup>2</sup> and currently in use in many ecosystems in Europe (Kainz et al., 2019).

<sup>1</sup>See <http://www.hubcap.eu>.

<sup>2</sup>See <http://midih.eu/>.

DIHIWARE offers a complete collaboration environment inspired by Enterprise Social Software (Cook, 2008). It supports both “Access to” and “Collaborate with” services, providing companies access to the latest knowledge, expertise and technology during their digital transformation paths toward piloting, testing and experimenting with new digital technologies.

The knowledge-driven services, complemented by the collaborative and innovation side of the platform, are intended to create a virtual environment where providers and consumers of digital technologies are not just matching assets and needs, but they are collaborating together towards joint innovations. This environment will be the foundation on which specific customisations (environment customisation, catalogue designing, and dedicated user journey) will be realised to meet the specific needs of HUBCAP.

The platform integrates open source technologies (e.g., those coming from the FIWARE Community<sup>3</sup> with enterprise-grade solutions such as Liferay Portal). DIHIWARE has four main subsystems (Figure 1a):

**Identity Manager:** This subsystem centralises user authentication, defining roles and granting access while using the other applications.

**Marketplace:** This subsystem handles the creation of company offerings by means of a product catalogue in which MBD assets and services will be shared. End users can view and interact with assets through this subsystem, while suppliers use it to manage their asset and service catalogue.

**Knowledge Base:** This subsystem supports semantic indexing and retrieval functions, grounded on semantic technologies and providing a set of services for creating awareness, dissemination, training, and managing connections among user-generated content.

**Social Portal:** This subsystem offers tools for social activity, user collaboration, matchmaking, and expert search, one of the key offerings on DIHIWARE.

HUBCAP extends this foundation framework with the sandbox capability. The overall idea is that it should form the centre of an ecosystem in which different organisations can undertake model-based collaborations. In order to protect intellectual property of the suppliers of assets, the platform needs to enable white-box, grey-box and black-box models so that it is possible to control access. This can for example be accomplished using the Functional Mockup Interface (FMI) standard that is supported by range of current tools (Blochwitz, 2014) to enable co-simulation

<sup>3</sup>See <http://fiware.org/>.

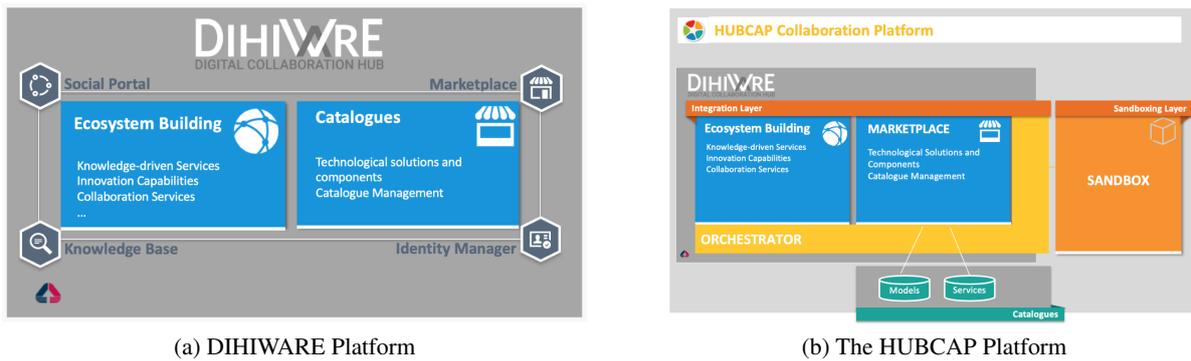


Figure 1: Growing from the DIHIWARE to the HUBCAP Platform.

of a collection of different individual models (Gomes et al., 2018).

The platform’s collaboration functions will enable diverse partners to work together through the sharing of controlled access to models. We imagine that an Original Equipment Manufacturer (OEM) may invite their suppliers to join the model-based development of a CPS. Many of the suppliers will wish to provide the models as black boxes. For the OEM this is not a problem as long these diverse models can be integrated in their analysis of CPS-level properties. The idea is that all the partners will be able collectively to analyse the composition from the sandbox.

### 3 THE SANDBOX FUNCTIONALITY

In a typical MBD setup there are three classes of assets:

- models, which are mathematical or formal abstractions of system elements (components or subsystems);
- tools, which are software packages and their dependencies that enable the development, analysis, and simulation of models; and
- Operating Systems (OSs), which refers to a software environment providing libraries and dependencies needed to run the tools.

Adding a sandbox feature to the DIHIWARE collaboration platform requires the support of a multitude of tools, dependencies, OSs. This calls for a virtualisation approach in which the tools involved in a given experiment are deployed for each individual user in an environment that is populated with all the dependencies required. In addition to plain virtualisation, a sandbox mechanism provides improved security and minimises perceived interference among users. We

envisage three kinds of sandboxes, each with a specific complexity and level of service:

**Demo Sandbox:** A simple web application demonstration, where inputs may be predefined for user experimentation. This supports simple cases in which a simple web site or application is sufficient to provide users with the inner workings of an experiment, serving as a potentially advanced/animated repository of the models and shared artefacts, with pointers on to the vendor sites. This serves as a solution for tool vendors/experiments with a need to host results without the need for significant dependence on the underlying OS features.

**Container Sandbox:** A web application deployed using a standard virtualised processes approach in which different containers host different services of the application and are connected through a virtual network interface. This option allows experiments involving tools requiring a single OS, but needing complex software environments (different versions of an environment) or a multiple services running in different containers.

**Cloud Sandbox:** A single sandbox is implemented by a set of virtual machines (VMs) operating on top of a Kernel-based Virtual Machine (KVM), connected through a KVM virtual network. The VMs inside a sandbox can access a common NFS storage to retrieve some input data (e.g., existing models) and to exchange data among themselves (e.g., generated models). This alternative supports experiments which are based on tools requiring different OSs or OS versions are allowed to run, exchange data via the network, and exchange files.

It is possible to realise a Demo or Container Sandbox using the Cloud type. A working prototype of this approach has been constructed, and details of it follow. The prototype allows a pre-registered HUB-

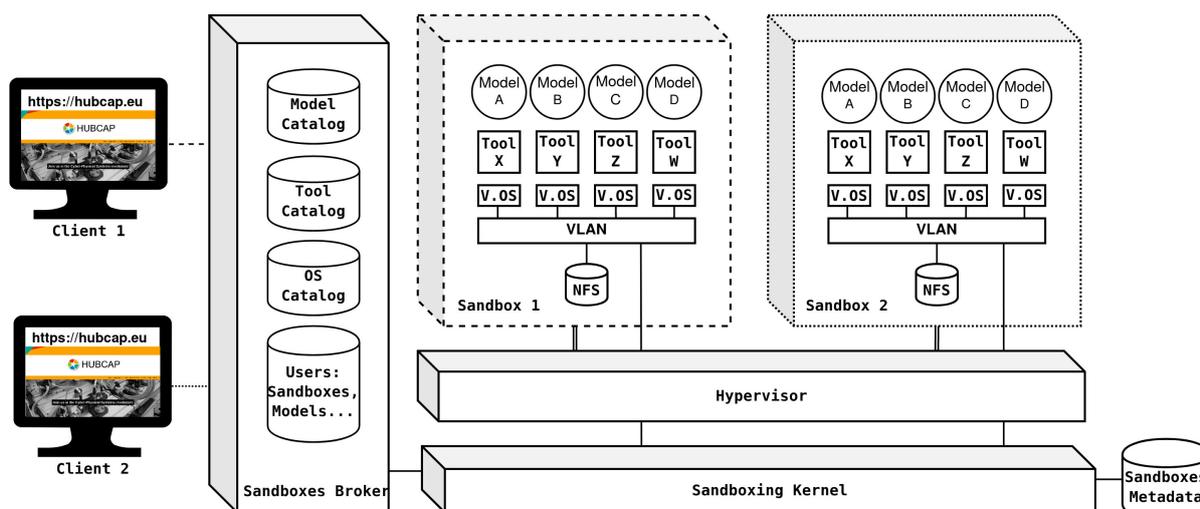


Figure 2: The HUBCAP Sandbox architecture.

CAP Platform user to select a combination of models, tools, and OS, pack them in an isolated sandbox and start playing with them via a web browser. Examples of use cases for the different kinds of sandboxes are given in Section 4.2.

### 3.1 Architecture

A sandbox is implemented as an isolated set of VMs (each one running a CPS tool) that interact with each other sharing a virtual dedicated subnet and a dedicated NFS storage service. No interaction is permitted between the VMs belonging to different sandboxes. The sandbox capability integrated with the DIHIWARE Platform is therefore a sort of private cloud service provider plus the middleware to manage and mediate the access to those cloud services. In addition, as many cloud service providers offer the capability to select a combination of hardware and operating systems, the HUBCAP Platform offers you to select a combination of OS environments, tools, and models to run an experiment using the HUBCAP sandbox feature.

The sandbox service is outlined in Figure 2. The DIHIWARE Platform is enhanced with a broker component (labelled as *Sandboxes Broker* in the figure), which hosts a web application mediating the access of different users (*Client 1* and *Client 2*) to the sandboxes they requested (*Sandbox 1* and *Sandbox 2* respectively). All the users will use an Internet browser to access the tools in the sandbox and all the interactions are mediated by the broker.

The *Sandbox Broker* has access to the catalogues of different models, tools, and pre-configured OSs that are available, so an end user can simply pick a valid combination to request a sandbox. In addition

to those catalogues, the *Sandbox Broker* keeps user information, such as the user's models (private copies of the model in the catalogue, which may have been modified by the user while using the sandbox) and the sandboxes the user created. This information is important to allow the creation of new sandboxes.

The operation of user requests and the sandboxing logic is provided by the *Sandboxing Kernel*, which is a component that interacts with the system *Hypervisor* to launch the different constituents of a sandbox, namely:

- *NFS* - Network File System providing storage in the form of shared folders where model files and tool outputs are placed.
- *VLANS* - Virtual networks restricting the communications of the VMs inside a sandbox to the set of VMs composing it and those only.
- *V.OS* - Virtual machines running the OSs supporting a tool, a remote desktop protocol to provide the clients access to the tool display, and other monitor and interoperability tools to operate the VM inside the Kernel.
- *Tools* - The tools running a model or a multi-model.
- *Models* - A mathematical/formal description of a component.

The operation relies on a database of metadata about the different sandboxes (the *Sandboxes Metadata* component in the figure). This component stores and keeps track of the sandboxes' states (running, suspended, ...) and user ownership of the resources. It is worth highlighting that the Kernel has direct network connections to the Sandboxes' VLANs.

### 3.2 Security

To ensure the data privacy of models and analysis results produced in the HUBCAP experiments, we envisage a security middleware to enhance the HUBCAP Platform with the due security and protection mechanisms. The sandbox design itself should ease security auditing and assurance. For instance, our current proposal follows a trusted “kernel architecture” where the sandboxes manager/broker launches sandboxes and mediates its interactions. An example of such is the multi-process architecture design of Google Chrome, where each tab containing untrusted code (not necessarily malicious, but not possible to be assumed secure) runs in a sandbox environment and accesses the system resources through a trusted broker ensuring independence of the different tabs. Thus it is possible to secure the platform by a general verification approach (Jomaa et al., 2018).

Moreover, the components of the sandbox kernel are open source, e.g. Hypervisor, NFS server, and the security will be based on Data-Service Sovereignty principles, in order to enhance trust among beneficiaries of the HUBCAP Platform (exchanging data and services), but also intended to provide protection mechanisms to prevent the infiltration of malware into the collaboration platform by applying known malware detection techniques, for instance (Macedo and Touili, 2013), which will systematically check the collaboration platform FMUs for malicious behaviour. Furthermore, secure isolation (Suciu et al., 2018) and security information and event management can ensure that aggregated data and log records can be automatically analysed giving a clear picture of what is happening on the platform.

### 3.3 Limitations

In the HUBCAP cloud environment, a single CPS tool is installed on a V.OS. Such a customised V.OS is then frozen and referenced in a tools catalogue, ready to be executed by the sandbox hypervisor by different HUBCAP users who are able pick up the same tool from the tools catalogue, run it on different sandboxes at the same time, and even customise it with other services and add them to the catalogue. This poses a problem when tool dependencies are not open-source software (OSS) or not resource efficient.

If an experiment demands a proprietary OS, which in turn demands a license per user, or blocks the possibility of using the OS in a virtualised hardware environment, the sandbox facility may be of limited value. We expect to explore and clarify these issues as tools are checked into the ecosystem and the analysis of

corner cases starts to provide results. Nevertheless, several issues need to be addressed before the sandboxing concept becomes a public service:

1. How to license sandboxes containing non-OSS or incompatible OSS licenses packages. The main question is whether the suppliers of such license-based software in the longer term will see an advantage also making versions of their software available in a pay-as-you-go servitisation approach.
2. How to deal with resource inefficient sandboxes. If the tool and model combination makes heavy use of resources (CPU, RAM, Storage) it may be difficult to obtain acceptable levels of performance in cases where this sandbox is made available for multiple users. A pay-as-you go solution approaches the issue using basic economics, but our research resources are limited and better use may be made of them in serving experiments that are ‘greener’ and allow more users to participate.

## 4 CATALOGUES OF MODELS AND MBD SERVICES

The HUBCAP Platform will provide access to assets including models and tools for MBD. The analysis capabilities of the tools will be available as services to be tested in a sandbox. The MBD services will include support for modelling of CPS with components, contracts, and equations, for analysis based on simulation, model checking, model-based safety analysis, for synthesis of HW/SW deployments, fault detection and recovery, planning, and for many more functionalities. We anticipate that the platform’s user community will integrate more tools and models over time.

Models and services will be presented to the user in catalogues, where the users will choose the tool, the kind of analysis they want to try, and existing models associated with it to exemplify the usage. The HUBCAP Platform will create a dedicated sandbox with the tool already installed and the desired models ready to be used. The user will follow the instructions to perform the analysis on the chosen model. The users will be able to write their own models and test the capabilities of the tool. If needed, the users will be able to get support by the tool experts via the collaboration services of the Platform.

### 4.1 Model Catalogue

We expect that users will supply adaptable and generic models to assist newcomers with specific

modelling tools and tool combinations. The Model Catalogue will support access to these. Initially, we expect to include models from standards and tutorials. For example, we expect to include all the public models that are available for the INTO-CPS tool chain (Mansfield et al., 2017) that can be imported and used directly inside the tools. Other example would include the wheel braking system architecture and the related fault trees described in the AIR6110 standard (S-18 Aircraft and Sys Dev and Safety Assessment Committee, 2011) and modelled in (Bozzano et al., 2015). A depiction of the current available models in a Cloud Sandbox prototype is shown in Figure 3.

## 4.2 MBD Services

In the following we provide examples of tools and services that may be realised using the HUBCAP Platform. To provide further detail in the conceptualisation of the sandbox service, we organise the examples by the kind of sandbox defined in Section 3.

**Example of a Demo Sandbox.** Controllab Products is a tool provider SME. Among other products that may be hosted in other kinds of sandboxes, they provide pay-as-you-go access to virtual reality 3D animations of model's simulations. The animations require high-end computer specifications and specialised VR hardware. In such cases hosting the experiments on a remote sandbox hosted on the HUBCAP Platform is neither feasible nor desired. Thus the sandbox may provide a simple web application with highlights of the full content and provide pointers to the replication of the experiment environment on an adequate hardware setup.

**The Container Example.** BEIA Consult is an SME that has developed models and a tool applied in energy efficiency of smart buildings (Suciu et al., 2019). The application is containerised using an open source platform, i.e. the tool is stored in containers/software packages that are run by a platform/container orchestration software such as Docker. Therefore, a sandbox with a single V.OS hosting the container orchestration software is available to execute the tool and models.

**The Cloud Example.** In this case a combination of Linux-based tools and Windows-only tools can be used to edit/display model details and generate Functional Mock-Units (FMUs). The FMUs may then be used to perform a co-simulation, the simulation of the behaviour of the joint executions of the different models. The co-simulation experiment may run in a third virtual machine hosting a containerised/cloud

version of the INTO-CPS co-simulation tool (Rasmussen et al., 2019). Although the INTO-CPS tool could be run in a single V.OS to perform a simulation of diverse models, the access to the edition and visualization of the models requires other tools, which is a feature that makes the INTO-CPS tool a good fit for the Cloud Sandbox service level.

## 5 OPEN CALLS

To encourage the population of the platform, and to evaluate its use as an aid to innovation in CPS design, HUBCAP will run a series of funded Open Calls. These will provide financial and technical support for SMEs to join the HUBCAP ecosystem and to experiment in highly innovative, cross-border experiments. There are three series of calls, each with different purposes:

**PULL Calls.** "PULL" calls encourage the population of the platform by model and tool suppliers. SMEs may request awards (up to €1,000) to aid in covering the costs of integration into the platform, including participation in a one-day workshop and 3 – 4 days overall effort. There will be five such calls and we expect to sponsor 200 projects.

**EXPERIMENT Calls.** "EXPERIMENT" calls will support consortia of typical two SMEs to experiment with the adoption of MBD for CPSs using assets and services from the platform, in particular from SMEs with less prior digital experience. Consortia may bid for €30,000 - €75,000 for projects of 4 to 6 months duration. There will be two **EXPERIMENT** calls and we expect to fund 20 to 30 projects.

**INNOVATE Call.** One **INNOVATE** call will fund up to €200,000 for consortia of SMEs to deploy new products and demonstrate highly innovative collaborations using the HUBCAP Platform. Funding supports a project of 12 months' duration. There will be one such call and we expect to grant 10 projects.

According to (De Prato et al., 2015), 63% of high potential innovations arise within projects in their final stages and 41% of all organisations behind these are SMEs. Therefore, as it is expected that new innovations will emerge towards the third year from the project third-party experiments, the **EXPERIMENT** and **INNOVATE** calls are supplied with a larger budget and are open later in the project timeline. Conversely, the **PULL** call is open from early in the project (with five regular deadlines) and has a smaller

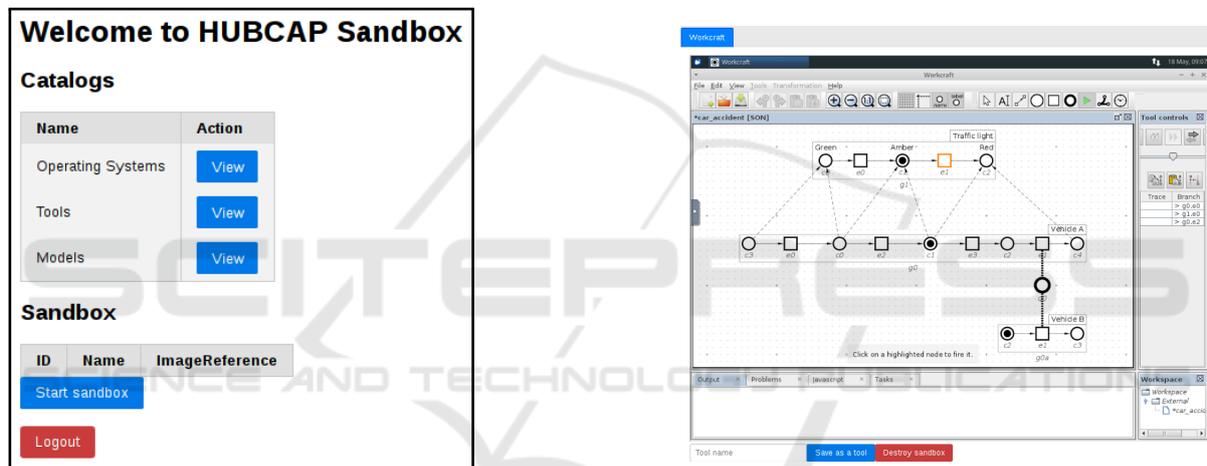
## Models Catalog

ID	name	description	actions	
13	SixBusPowerSystem	SixBusPowerSystem under variable load	<a href="#">Details</a>	<a href="#">Delete</a>
14	RBTS	RBTS	<a href="#">Details</a>	<a href="#">Delete</a>
15	SakilaDB	SakilaDB	<a href="#">Details</a>	<a href="#">Delete</a>
16	INTO-CPS Tutorial 1	Experiments First Tutorial	<a href="#">Details</a>	<a href="#">Delete</a>
18	Workcraft_BasicVendingMachine	Vending machine finite state machine	<a href="#">Details</a>	<a href="#">Delete</a>

[Upload New Model](#)

[Go Back](#)

Figure 3: Depiction of already available HUBCAP Platform models, which are presented to a user of the Cloud Sandbox prototype developed so far.



(a) The landing page presented to a user by the current prototype of the Cloud Sandbox.

(b) A Cloud Sandbox loaded with a tool and a Workcraft Petri net model running.

Figure 4: Snapshots from the current Cloud Sandbox prototype.

budget dedicated to workshops helping partners get their assets into the collaboration platform.

## 6 CONCLUDING REMARKS AND FUTURE WORK

The HUBCAP Platform is under development, and we expect that the first public version will become available in late 2020. Initial snapshots of its sandboxing prototype are available (Figure 4). The platform is intended to form a shared resource for an ecosystem in which diverse organisations will supply models and tools to encourage and ease the evaluation and adoption of MBD approaches for CPSs.

Our hope is that the HUBCAP ecosystem sup-

ported by this platform might encourage development of MBD through servitisation, enabling users and tool suppliers to explore, share, and buy CPS assets (models, tools, services, training) from across the ecosystem through a “test-before-invest” sandbox and – at least in some cases – integrated “pay-as-you-go” charging.

We anticipate that, in the course of populating the collaboration platform, we will run into limitations in the capabilities of both tools and the sandbox architecture. It is envisaged that there will be challenges in regards to both licenses for OSs as well as tools with special needs for particular hardware support (e.g., graphics cards) that may not easily be supported in a sandbox context.

Despite the challenges, we hope that the HUBCAP Platform will be extended in several directions

enabling true collaboration between different participating organisations, alongside and as a result of the open calls. We envisage that the HUBCAP Platform may be conveniently hosted at standard cloud operators as well as on servers at large companies with many suppliers such that they can be in full control of the development of the collaboration around larger CPSs such as automobiles and airplanes.

## ACKNOWLEDGEMENTS

The work presented here is partially supported by the HUBCAP Innovation Action funded by the European Commission's Horizon 2020 Programme under Grant Agreement 872698. We would also like to thank Giuseppe Veneziano for his input and snapshots of the Cloud Sandbox prototype, and Nick Battle, Claudio Gomes and the reviewers for their comments on drafts of this paper.

## REFERENCES

- Beckers, J., Heemels, M., Bukkems, B., and Muller, G. (2007). Effective industrial modeling: The example of happy flow. In Heemels, M. and Muller, G., editors, *Boderc: Model-based design of high-tech systems*, chapter 6, pages 77–88. Embedded Systems Institute, Den Dolech 2, Eindhoven, The Netherlands, second edition.
- Blochwitz, T. (2014). Functional Mock-up Interface for Model Exchange and Co-Simulation. <https://www.fmi-standard.org/downloads>.
- Bozzano, M., Cimatti, A., Pires, A. F., Jones, D., Kimberly, G., Petri, T., Robinson, R., and Tonetta, S. (2015). Formal Design and Safety Analysis of AIR6110 Wheel Brake System. In *CAV*, pages 518–535.
- Cook, N. (2008). *Enterprise 2.0: How social software will change the future of work*. Gower Publishing Limited.
- De Prato, G., Nepelski, D., Piroli, G., et al. (2015). Innovation Radar: Identifying Innovations and Innovators with High Potential in ICT FP7, CIP & H2020 Projects. *Science and Policy Report, Joint Research Centre*.
- Fritzson, P. (2015). *Principles of Object-Oriented Modeling and Simulation with Modelica 3.3: A Cyber-Physical Approach*. IEEE Press. Wiley, 2 edition.
- Gomes, C., Thule, C., Broman, D., Larsen, P. G., and Vangheluwe, H. (2018). Co-simulation: a Survey. *ACM Comput. Surv.*, 51(3):49:1–49:33.
- Jomaa, N., Nowak, D., Grimaud, G., and Hym, S. (2018). Formal proof of dynamic memory isolation based on mmu. *Science of Computer Programming*, 162:76–92.
- Kainz, O., Jakab, F., Michalko, M., Hudák, M., and Petija, R. (2019). Enhanced approaches to automated monitoring environmental quality in non-isolated thermodynamic system. *IFAC-PapersOnLine*, 52(27):365–376.
- Larsen, P. G., Fitzgerald, J., Woodcock, J., Fritzson, P., Brauer, J., Kleijn, C., Lecomte, T., Pfeil, M., Green, O., Basagiannis, S., and Sadovykh, A. (2016). Integrated Tool Chain for Model-based Design of Cyber-Physical Systems: The INTO-CPS Project. In *CPS Data Workshop*, Vienna, Austria.
- Macedo, H. D. and Touili, T. (2013). Mining malware specifications through static reachability analysis. In *European Symposium on Research in Computer Security*, pages 517–535, Berlin, Heidelberg. Springer, Springer Berlin Heidelberg.
- Mansfield, M., Gamble, C., Pierce, K., Fitzgerald, J., Foster, S., Thule, C., and Nilsson, R. (2017). Examples compendium 3. Technical report, The INTO-CPS Project.
- Rasmussen, M. B., Thule, C., Macedo, H. D., and Larsen, P. G. (2019). Migrating the INTO-CPS Application to the Cloud. In Gamble, C. and Couto, L. D., editors, *The 17th Overture workshop*, pages 47–61, Porto, Portugal. Newcastle University Technical Report Series. CS-TR-1530.
- S-18 Aircraft and Sys Dev and Safety Assessment Committee (2011). AIR 6110, Contiguous Aircraft/System Development Process Example. Standard, SAE International.
- Suciu, G., Istrate, C., Petrache, A., Schlachet, D., and Buteau, T. (2018). On demand secure isolation using security models for different system management platforms. In *Advanced Topics in Optoelectronics, Microelectronics, and Nanotechnologies IX*, volume 10977, page 109770R. International Society for Optics and Photonics.
- Suciu, G., Necula, L., Iosu, R., Usurelu, T., and Ceaparu, M. (2019). Iot and cloud-based energy monitoring and simulation platform. In *2019 11th International Symposium on Advanced Topics in Electrical Engineering (ATEE)*, pages 1–4. IEEE.
- Sztipanovits, J., Bapty, T., Neema, S., Koutsoukos, X., and Jackson, E. (2015). Design Tool Chain for Cyber-physical Systems: Lessons Learned. In *Proceedings of the 52nd Annual Design Automation Conference, DAC '15*, pages 81:1–81:6, New York, NY, USA. ACM.
- Van der Auweraer, H., Anthonis, J., De Bruyne, S., and Leuridan, J. (2013). Virtual engineering at work: the challenges for designing mechatronic products. *Engineering with Computers*, 29(3):389–408.