A Framework for Cyber-Physical Systems Design – A Concept Study

Ondrej Rysavy
Faculty of Information Technology
Brno University of Technology
Czech Republic
E-mail: rysavy@fit.vutbr.cz

Miroslav Sveda
Faculty of Information Technology
Brno University of Technology
Czech Republic
E-mail: sveda@fit.vutbr.cz

Radimir Vrba
Faculty of Electrical Eng.& Comm.
Brno University of Technology
Czech Republic
E-mail: vrbar@feec.vutbr.cz

Abstract — **The paper deals with principles of a launching research focused on cyber-physical systems (CPS) design environment. It refers to completed real-world CPS application projects aimed at smart data acquisition systems capable to store and present measured data wirelessly. The paper depicts a CPS design approach stemming from generic requirements on domain applications, reviews state of the art of the domain, mentions some experience with pilot projects and brings an outline of the intended CPS design framework.**

Keywords- Embedded system design, smart sensor, wireless communication, temperature and pressure measurement.

I. INTRODUCTION

The term Cyber-Physical Systems has come to describe the research and technological effort that will ultimately efficiently allow interlinking the real world physical objects and cyberspace. The integration of physical processes and computing is not new. Embedded systems have been in place since a long time to denote systems that combine physical processes with computing. The revolution is coming from communicating embedded computing devices that will allow instrumenting the physical world with pervasive networks of sensor-rich, embedded computation.

This paper deals with principles of a launching research focused on CPS design environment. It refers to completed real-world application projects aimed at smart data acquisition systems capable to store and present measured data wirelessly. The paper depicts a CPS design approach stemming from generic requirements on domain applications, state-of-the-art review, and discussing initial ideas on the proposed design and development tools interconnected to form a development environment.

II. REQUIREMENTS

The CPS research program aims to develop a unifying theory for the design and implementation of integrated cyber and physical resources that can be applied across multiple domains [12]: "... Currently, unrelated methods are used to separately develop cyber and physical subsystems. The differences between the two sides are manifest at the most fundamental levels: computer science builds upon discrete mathematics, whereas engineering is dominated by continuous mathematics. Even within the broad fields of engineering and computer science, multiple sub-disciplines use dissimilar concepts and tools. The lack of a unifying or composable theory makes it impossible to guarantee safety and performance by design. System validation requires extensive testing — an approach that is becoming intractable as systems become more complex. Despite progress in the development of increasingly more powerful technologies for networked embedded sensing and control, today's embedded computing systems are point solutions for specific applications. Current approach to hardware and software design, systems engineering and real-time control needs to be rethought in the unifying context of cyber-physical systems. For example, open, flexible, and extensible architectures for cyber-physical systems would enable and better influence advances in hardware and software components and subsystems. They should be analyzable and should support principled composition or integration. Run-time operation should exploit information-rich environments to enhance performance and reliability. In some applications, these systems should be context aware, with the ability to modify their behaviors to accommodate changing configurations, adapt to variations in the environment, sustain safe operation, and improve performance over time. For many applications, cyber-physical systems must be certifiable, i.e. new approaches are needed for the specification, verification, and validation of tightly integrated cyber and physical elements."

As is generally agreed, the effective design of cyber-physical systems requires research advances in methods and tools that support multiple views of integrated cyber and physical components. New programming languages are needed to handle complex interactions between cyber and physical resources and to deal with unstructured data and stringent requirements for responsiveness. Algorithms for reasoning about and formally verifying properties of complex integrations of cyber and physical resources are needed. Tools for implementing algorithms to support off-line and run-time optimization and control are also needed. Tools should support concurrent engineering of physical systems with sensing, communication, and control architectures. Methods and tools should enable new forms of analysis, testing, and validation of integrated discrete and continuous dynamics at multiple temporal and spatial scales and different levels of resolution. Tools should be open, interoperable, and highly expressive to enhance productivity and enable community use. They should also be extensible to leverage new results from the foundations research and

 ISBN: 978-1-61208-184-7

accommodate new technologies and capabilities as they become available.

III. STATE OF THE ART

Many of the embedded systems-related studies and efforts in the past have focused on the challenges the physical environment brings to the scientific foundations of networking and information technology, see [2] and [4]. However, the full scope of the change enabled by introducing CPS as a new branch of science and technology provides much more than restructuring inside this domain. The new approach can turn entire industrial sectors into producers of CPS. Actually, CPS is about merging computing and networking with physical systems to create new capabilities and improve product quality [11].

Cyber-physical systems denote a new modeling paradigm that promotes a holistic view on real-world – and therefore complex – systems. These systems have been studied before from various particular perspectives using paradigms like ubiquitous and distributed computing or embedded and hybrid systems. The above mentioned facts require also another approach to the design of such systems respecting from the beginning of design process the application domain that influences quality-of-service requirements such as real-time behavior, safety and security [17], [18], [14] and [15], but also precision, reliability and other non-functional properties affecting attributes specified usually by official standards [9].

In a CPS application, the function of a computation is defined by its effect on the physical world, which is in this case not only a system environment, but evidently also a component of the designed application system. Therefore, proper design environments should be used to improve or at least to enable efficiency of the design process. In cyber-physical systems the passage of time becomes a central feature — in fact, it is this key constraint that distinguishes these systems from distributed computing in general. Time is central to predicting, measuring, and controlling properties of the physical world: given a (deterministic) physical model, the initial state, the inputs, and the amount of time elapsed, one can compute the current state of the plant. This principle provides the foundations of control theory. However, for current mainstream programming paradigms, given the source code, the program's initial state, and the amount of time elapsed, we cannot reliably predict future program state. When that program is integrated into a system with physical dynamics, this makes principled design of the entire system difficult. Instead, engineers are stuck with a prototype-and-test style of design, which leads to brittle systems that do not easily evolve to handle small changes in operating conditions and hardware platforms. Moreover, the disparity between the dynamics of the physical plant and the program seeking to control it potentially leads to errors, some of which can be catastrophic.

IV. DESIGN FRAMEWORK

Perri and Kaiser [13] formulate a model of development environment employing three tool types: (1) structures in the role of reusable components embodied into developed systems; (2) mechanisms in the role of proper development tools used for development process but not included into developed systems, and (3) strategies as design and development methods. The design framework's concepts can stem from verifiable formal specifications of CPS as a launching paradigm and from reusability paradigm supporting all phases of the design process ranging from specification to implementation and testing.

The basic terms, excerpted originally from [24] and adapted according to [21] for computer-based systems and, particularly, for embedded systems application area purpose can be restated as follows:

- The environment is the portion of a real world relevant to the design project.
- The embedded system is a computer-based artifact that will be constructed and connected to the environment, as a result of the design project.
- A requirement is an embedded system's property intended to express the desires of the customer concerning the design project.
- A statement of domain knowledge is an environment's property intended to be relevant to the design project.
- A specification is an embedded system's property, intended to be directly implementable and to support satisfaction of the requirements.

Let S be the set of specifications, R be the set of requirements, and K be the relevant domain knowledge for a design project. Then S and K must be sufficient to guarantee that the requirements R are satisfied. The primary role of domain knowledge, K, is to bridge the gap between requirements, R, and specifications, S. Requirements that are not specifications are always converted into specifications with the help of domain knowledge. Application patterns, which embody domain knowledge, deal both with specifications and implementations.

It would seem, that the framework can be with no trouble reformulated for CPS and, after that, it can offer a starting point for specification and design. Evidently, such framework has to be refined to be useful for real applications. On the other hand, it should be general-enough to support broad application domains. The next section presents some experience based on completed CPS applications developed without special design environment, but demonstrating typical design cases.

V. LESSONS LEARNED

Starting with [23], [8], we collected some experience with designs of deeply imbedded CPS in frame of the following research outcomes: (1) mobile temperature data logger based on RFID system [17], (2) optoelectronic pressure and temperature sensory system based on dedicated Bluetooth network [17], and (3) optoelectronic pressure sensor system based on distributed architecture with Intranet TCP/IP [18]. After reviewing basic design concepts deployed, main attention is focused in all cases on the CPS artifacts' specification, design with respect to application

 ISBN: 978-1-61208-184-7

domain requirements, assembly, and appropriate communication services.

The paper [17] describes two CPS designs in more detail using two original research outcomes: mobile temperature data logger based on RFID system, and optoelectronic pressure sensory system based on dedicated Bluetooth network. The presented temperature data logger stands for an example of flexible, mobile and intelligent appliances fitting various industrial or medical applications. Similarly, the discussed sensor network represents a system architecture stemming from wireless smart pressure sensors connected by Bluetooth and from a network concentrator, which is based either on PDA personal digital assistant or on GSM SmartPhone.

The paper [18] describes a CPS example using an optoelectronic pressure sensor system based on distributed architecture with Internet/Intranet TCP/IP structure exploiting Ethernet 10/100 Mbps. After reviewing basic CPS concepts deployed, main attention is focused on a concrete optoelectronic pressure sensor design, assembly, and communication services in frame of the multi-sensor system fitting the application requirements. The networking configuration exemplifies in this case a real solution of a more complex networked embedded system application based on the IEEE 1451 family of standards and on actual software and hardware components developed by the authors and collaborators for a class of sensing and measurement embedded applications.

Both above mentioned papers strive to demonstrate application-driven designs of deeply embedded CPS cases that differ substantially in technology used. On the other hand, those cases enable also to identify typical commonalities in detailed CPS designs and, hence, to derive general requirements on design tools.

VI. CYBER-PHYSICAL SYSTEMS DESICN CONCEPTION

Design and development systems, see e.g. [4], [9], [5], [7], [22], have to support important concepts and methods by their tools for complete design and development life cycle of applications belonging to considered application domains. The toolset related to the discussed design framework will necessarily include also original methods and tools. At the beginning, the development means will target predominantly front-end parts of specification and design, namely formal specification, verification and rapid prototyping.

Conventional verification techniques to be used in the development environment have high memory requirements and are very computationally intensive. Therefore, they are unsuitable for real-world CPS systems that exhibit complex behaviors and cannot be efficiently handled unless we use scalable methods and techniques [20], which exploit fully the capabilities of new hardware architectures and software platforms [8]. High-performance verification techniques focus on increasing the amount of available computational power. These are, for example, techniques to fight memory limits with efficient utilization of external techniques that introduce cluster-based algorithms to employ aggregate power of network-interconnected computers, or techniques to speed-up the verification on multi-core processors.

Researching CPS models consist of capturing characteristics of CPS. We plan to study existing and to propose new models for common architectural and behavioral artifacts and communication patterns of the CPS domain.

To be more explicit, at the beginning we define models using Ptolemy II framework (see [14], [10]) extended by existing formal tools and we will study the possibility to integrate the formal verification methods for these hybrid models. It would require examining carefully the semantics bound in different models and define precise transformations to extract verifiable models from design models.

Domain specific modeling languages (DSML), contrary to the universal modeling languages, are specifically customized to the area of problems being solved. Using DSML approach, the modeling of a system is itself preceded by the phase of meta-modeling of the application domain. We plan to propose a DSML for the reliable real-time embedded devices in smart sensor and control networks domain and provide formal semantics for this language that should enable applications of formal methods for transformation and verification of CPS properties.

We will research possibility to apply existing formal methods to the models generated from the specifications written in CPS-DSML. The models describe the system being developed at different levels and views. Automated tools should support inter-model validation. Thus our primary concern is to demonstrate how tools based on formal methods can proof the inter-model consistency and property preservation. For instance, model of software components, which behavior is driven by discrete means of computing should be in consistency with lower level model of hardware processing units and also with same level model of abstract environment behavior. The difficulty and novelty lies in consideration that different models obey different means of computing.

Designed development environment prototype will include tools and methods that can be used to approach demonstration and experimenting with the selected application area. We assume that various methods will be experimentally implemented as software tools to show the capability of the approach on non-trivial use cases. New design patterns and components will be created and verified in frame of case studies. These case studies will serve to gather experience in development of CPS. The work should conclude by critical evaluation of the proposed approach, showing the strength aspects of considered method and revealing drawbacks that deserve further research.

VII. CONCLUSIONS

The paper deals with principles of a launching research focused on CPS design environment. It refers to completed real-world application projects aimed at smart data acquisition systems capable to store and present measured data wirelessly. The paper depicts a CPS design approach stemming from generic requirements on domain applications, continuing with brief review on state-of-the-art, and completed by initial ideas on the proposed design and

 ISBN: 978-1-61208-184-7

development tools interconnected to form a development environment.

This paper utilizes as model demonstrations three CPS designs rooted in original research outcomes: mobile temperature data logger based on RFID system, optoelectronic pressure sensory system based on dedicated Bluetooth network, and optoelectronic pressure sensory system based on Ethernet/IP/TCP network, published in more detail in frame of previous ICONS conferences.

Our research group is currently launching a related continuation research that aims at the formal tools support of CPS design [17], [18], [19]. Evidently, this new research domain requires not only formal specification and verification techniques extensions and modifications, but also novel approaches and adaptations of such general methods as model checking and proving, see e.g. [1], [2], [3], [9], [10], [16] and [22].

ACKNOWLEDGMENT

This project has been carried out with a financial support from the Czech Republic state budget through the *IT4Innovations Centre of Excellence*, EU, CZ 1.05/1.1.00/02.0070CEZ and through the MMT project no. MSM0021630528: *Security-Oriented Research in Information Technology*, by the Technological Agency of the Czech Republic through the grant no. TA01010632: *SCADA system for control and monitoring RT processes*, and by the Brno University of Technology, Faculty of Information Technology through the specific research grant no. FIT-S-11-1: *Advanced Secured, Reliable and Adaptive IT*. We also strive for the support by the Grant Agency of the Czech Republic through the grant proposal *Designing Cyber-Physical Systems.*

REFERENCES

[1] R. Akella and B.M. McMillin, Model-checking BNDC Properties in Cyber-Physical Systems, *Proceedings of the33rd INternational Computer Software and Applications Conference COMPSAC 2009*, IEEE CS, New York, NY, US, 2009, pp.660-663.

[2] B. Bonakdarpour, Challenges in Transformation of Existing Real-Time Embedded Systems to Cyber-Physical Systems *IEEE Symposium on Real-Time Systems RTSS RTSS - Ph.D. Forum on Deeply Real-Time Embedded Systems*, Tucson, Arizona, 2007, 2pp.

[3] M.C. Bujorianu and H.Barringer, An Integrated Specification Logic for Cyber-Physical Systems, *Proceedings of the 14th IEEE International Conference on Engineering of Complex Computer Systems*, Potsdam, Germany, 2009, pp.91-100.

[4] J. C. Eidson, E.A. Lee, S. Matic, S.A. Seshia and J. Zou, Time-centric Models For Designing Embedded Cyber-physical Systems, EECS Department, University of California, Berkeley, *Technical Report No. UCB/EECS-2009-135*, October 9, 2009.

[5] E.K. Jackson and J. Sztipanovits, Correct-ed through Construction: A Model-based Approach to Embedded Systems Reality. *Proceedings of the 13th Engineering of Computer-Based Systems*, IEEE Computer Society, Los Alamitos, CA, pp.164-173, 2006.

[6] J.E. Kim and Daniel Mosse, Generic framework for design, modeling and simulation of cyber physical systems, *SIGBED Review*, Vol. 5, No. 1, ACM, January 2008, 2pp.

[7] B.H. Krogh, E. Lee, I. Lee, A. Mok, R. Rajkumar, L.R. Sha, A.S. Vincentelli, K. Shin, J. Stankovic, J. Sztipanovits, W. Wolf and W. Zhao, *Cyber-Physical Systems, Executive Summary,* CPS Steering Group, Washington D.C., March 6, 2008. [http://www.nsf.gov/pubs/2008/nsf08611/nsf08611.htm]

[8] R. Kuchta, P. Steffan, Z. Barton, R Vrba and M. Sveda, Wireless Temperature Data Logger, *Proceedings of the 2005 Asian Conference on Sensors, and International Conference on new Techniques in Pharmaceutical and Biomedical Research*, 5-7 Sept. 2005, pp.208-212.

[9] E.A. Lee, Computing Needs Time, *Communications of the ACM*, Vol.52, No.5, pp.70-79, May 2009.

[10] E.A. Lee. *Finite State Machines and Modal Models in Ptolemy II*, Technical report, EECS Department, University of California, Berkeley, UCB/EECS-2009-151, December, 2009.

[11] E.A. Lee, CPS Foundations, *Proceedings of the DAC'10,* ACM, Anaheim, California, June 2010, pp.737-742.

[12] National Science Foundation, *Cyber-Physical Systems Program Solicitation, NSF 10-515*, Arlington, VA, US, March 11, 2010

[13] D.E. Perri and G.E. Kaiser, Models of Software Development Environment, *IEEE Transactions on Software Engineering*, Vol.17, 1991, pp.283-295.

[14] PtolemyII: http://ptolemy.berkeley.edu/ptolemyII

[15] L. Sha and J. Meseguer, Design of Complex Cyber Physical Systems with Formalized Architectural Patterns, *Software-Intensive Systems and New Computing Paradigms: Challenges and Visions,* Springer, 2008, pp 92-100.

[16] J.A. Stankovic, I. Lee, A. Mok and R. Rajkumar, Opportunities and obligations for physical computing systems, *IEEE Computer*, November 2005*,* pp.23-31.

[17] M. Sveda and R. Vrba, A Cyber-Physical System Design Approach, *Proceedings of The Sixth International Conference on Systems - ICONS 2011*, St. Maarten, AN, IARIA, 2011, pp.12-18.

[18] M. Sveda and R. Vrba, An Embedded Application Regarded as Cyber-Physical System, *Proceedings of the Fifth International Conference on Systems ICONS 2010*, Les Menuires, FR, IARIA, 2010, pp.170-174.

[19] M. Sveda and R. Vrba, Meta-Design with Safe and Secure Embedded System Networking, *International Journal On Advances in Security*, Vol. 2, No. 1, 2009, US, pp.8-15.

[20] M. Sveda and R. Vrba, Specifications of Secure and Safe Embedded System Networks, *8th International Conference on Networks Proceedings ICN 2009*, New York, NY, US, IARIA, IEEE CS, 2009, pp.220-225.

[21] M. Sveda, A Design Framework for Internet-Based Embedded Distributed Systems, *Proceedings of the International IEEE Conference and Workshop ECBS'2004*, Brno, Czech Republic, IEEE Computer Society Press 2004, pp.113-120.

[22] H. Tang and B.M. McMillin, Security Property Violation in CPS through Timing, *Proceedings of the 28th on Distributed Computing Systems IDCS 2008, Workshops*, IEEE CS, New York, NY, US, 2008, pp.519-524.

[23] R. Vrba, O. Sajdl, R. Kuchta and M. Sveda., Wireless Smart Sensor Network System. *Proceedings of the ICSE & INCOSE. Conference*, Las Vegas, Nevada: CRC Press LLC, 2004. pp.104-109.

[24] P. Zave and M. Jackson, Four Dark Corners of Require-ments Engineering, *ACM Transactions on Software Engineering and Methodology*, Vol.6, No.1, 1997, pp.1-30.

 ISBN: 978-1-61208-184-7