

# Semantics of Cyber-Physical Systems

Tharam Dillon<sup>1</sup>, Elizabeth Chang<sup>2</sup>, Jaipal Singh<sup>3</sup>, and Omar Hussain<sup>2</sup>

<sup>1</sup> La Trobe University, Australia

tharam.dillon7@gmail.com

<sup>2</sup> School of Information Systems, Curtin University, Australia

{Elizabeth.Chang,O.Hussain}@curtin.edu.au

<sup>3</sup> Dept. of Electrical and Computer Engineering, Curtin University, Australia

J.Singh@curtin.edu.au

**Abstract.** The very recent development of Cyber-Physical Systems (CPS) provides a smart infrastructure connecting abstract computational artifacts with the physical world. The solution to CPS must transcend the boundary between the cyber world and the physical world by providing integrated models addressing issues from both worlds simultaneously. This needs new theories, conceptual frameworks and engineering practice. In this paper, we set out the key requirements that must be met by CPS systems, review and evaluate the progress that has been made in the development of theory, conceptual frameworks and practical applications. We then discuss the need for semantics and a proposed approach for addressing this. Grand challenges to informatics posed by CPS are raised in the paper.

## 1 Introduction

The very recent development of Cyber-Physical Systems (CPS) provides a unified framework connecting the cyber world with the physical world. CPS allows for robust and flexible systems with multi-scale dynamics and integrated wired and wireless networking for managing the flows of mass, energy, and information in a coherent way through integration of computing and communication capabilities with the monitoring and/or control of entities in the physical world in a dependable, safe, secure, efficient and real-time fashion.

CPS has recently been listed as the No.1 research priority by the U.S. President's Council of Advisors on Science and Technology [2]. This led the US National Science Foundation to organize a series of workshops on CPS [3]. The CPS framework has the capability to tackle numerous scientific, social and economic issues. The three applications for CPS are in future distributed energy systems, future transportation systems and future health care systems [1,4,14]. These applications will require seamless and synergetic integration between sensing, computation, and control with physical devices and processes.

Building CPS is not a trivial task. It requires a new ground-breaking theory that models cyber and physical resources in a unified framework. This is a huge challenge that none of the current state-of-the-art methods are able to overcome due to the fact that computer science and control theory are independently developed based on

overly-simplified assumptions of each other. For example, many key requirements (e.g. uncertainty, inaccuracy, etc.) crucial to physical systems are not captured and fully dealt with in the computer science research agenda. In a similar vein, computational complexity, system evolution and software failure are often ignored from the physical control theory viewpoint, which treats computation as a precise, error-free, static 'black-box'. The solution to CPS must transcend the boundary between the cyber world and the physical world by providing a unified infrastructure that permits integrated models addressing issues from both worlds simultaneously. This paper begins by setting out the requirements and then evaluates the progress that has been made in addressing these [2]. This paper will show that CPS calls for new theories and some fundamental changes to the existing computing paradigm [4-6].

In particular it will look at the need for semantics and propose a framework for such semantics.

## 2 Requirements of CPS

The National Science Foundation (NSF) CPS Summit held in April 2008 [4] defines CPS as "physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core". Researchers from multiple disciplines such as embedded systems and sensor networks have been actively involved in this emerging area.

Our vision of CPS is as follows: networked information systems that are tightly coupled with the physical process and environment through a massive number of geographically distributed devices [1]. As networked information systems, CPS involves computation, human activities, and automated decision making enabled by information and communication technology. More importantly, these computation, human activities and intelligent decisions are aimed at monitoring, controlling and integrating physical processes and environment to support operations and management in the physical world. The scale of such information systems range from micro-level, embedded systems to ultra-large systems of systems. Devices provide the basic interface between the cyber world and the physical one.

The discussions in the NSF Summit [4] can be summarized into eleven scientific and technological challenges for CPS solutions. These challenges constitute the top requirements for building cyber-physical systems and are listed below: (1) Compositionality (2) Distributed Sensing, Computation and Control; (3) Physical Interfaces and Integration; (4) Human Interfaces and Integration; (5) Information: From Data to Knowledge; (6) Modeling and Analysis: Heterogeneity, Scales, Views; (7) Privacy, Trust, Security; (8) Robustness, Adaptation, Reconfiguration; (9) Software; (10) Verification, Testing and Certification; (11) Societal Impact.

Based on the challenges listed above, a new unified cyber-physical systems foundation that goes beyond current computer mediated systems needs to be developed. We explain how this can be achieved, in-line with the challenges to CPS identified by the NSF summit report.

CPS need to support resource composition that integrates heterogeneous components from both the physical and cyber world. This requires: (1) a "plug-n-play" framework that provides a high degree of adaptivity and (re-) programmability

supported by flexible interface definitions and integration styles (e.g. synchronous, asynchronous, continuous, and discrete) and (2) the capability to model and predict run-time performance and behavior of the evolving composite.

CPS need to make distributed control, sensing, and communication. This requires: (1) information collection and decision making in a distributed network environment with high latency and high uncertainty, (2) new theories are needed on the use of network communication at the micro-level to prevent adverse effects at the macro-scale in a distributed environment, (3) new theories are needed on control without centralized administration, measurement and perhaps incomplete information.

CPS need to stay in constant touch with physical objects. This requires: (1) models that abstract physical objects with varying levels of resolutions, dimensions, and measurement scales, (2) mathematical representation of these models and understanding of algorithmic, asymptotic behavior of these mathematical models, and (3) abstractions that captures the relationships between physical objects and CPS.

Humans have to play an essential role (e.g. influence, perception, monitoring, etc.) in CPS. This requires: (1) seamless integration and adaptation between human scales and physical system scales. (2) support for local contextual actions pertinent to specific users, who are part of the system rather than just being the "users" of the system, (3) new theories on the boundary (e.g. hand-over or switch) between human control and (semi-) automatic control.

Many CPS are aimed at developing useful knowledge from raw data. This requires (1) algorithms for sensor data fusion that also deal with data cleansing, filtering, validation, etc. (2) data stream mining in real-time (3) storage and maintenance of different representations of the same data for efficient and effective (e.g. visualization) information retrieval and knowledge extraction.

CPS needs to deal with massive heterogeneity when integrating components of different natures from different sources. This requires (1) integration of temporal, event-driven, and spatial data defined in significantly different models (asynchronous vs. synchronous) and scales (e.g. discrete vs. continuous), (2) new computation models that characterize dimensions of physical objects such as time (e.g. to meet real-time deadline), location, energy, memory footprint, cost, uncertainty from sensor data, etc., (3) new abstractions and models for cyber-physical control that can deal with - through compensation, feedback processing, verification, etc. - uncertainty that is explicitly represented in the model as a "first-class citizen" in CPS, (4) new theories on "design for imperfection" exhibited by both physical and cyber objects in order to ensure stability, reliability, and predictability of CPS, (5) system evolution in which requirements and constraints are constantly changing and need to be integrated into different views of CPS, and (6) new models for dealing with issues in large-scaled systems such as efficiency trade-offs between local and global, emergent behavior of complex systems, etc.

CPS in general reveal a lot of physical information, create a lot of data concerning security (e.g. new types of attacks), privacy (e.g. location), and trust (e.g. heterogeneous resources). This requires: (1) new theories and methods on design principles for resilient CPS, threat/hazard analysis, cyber-physical inter-dependence anatomy,

investigation/prediction of gaming plots at different layers of CPS, (2) formal models for privacy specification that allow reasoning about and proof of privacy properties, (3) new mathematical theories on information hiding for real-time streams, (4) light-weight security solutions that work well under extremely limited computational resources (e.g. devices), (5) new theories on confidence and trust maps, context-dependent trust models, and truth/falseness detection capabilities.

Due to the unpredictability in the physical world, CPS will not be operating in a controlled environment, and must be robust to unexpected conditions and adaptable to subsystem failures. This requires: (1) new concepts of robust system design that deals with and lives on unexpected uncertainties (of network topology, data, system, etc.) occurring in both cyber and physical worlds, (2) the ability to adapt to faults through (self-) reconfiguration at both physical and cyber levels, (3) fault recovery techniques using the most appropriate strategies that have been identified, categorized, and selected, (4) system evolution through learning faults and dealing with uncertainties in the past scenarios, (5) system evolution through run-time reconfiguration and hot deployment.

One important omission from the above requirements is the need for semantics. In particular semantics that are capable of bridging the real physical world and the virtual world.

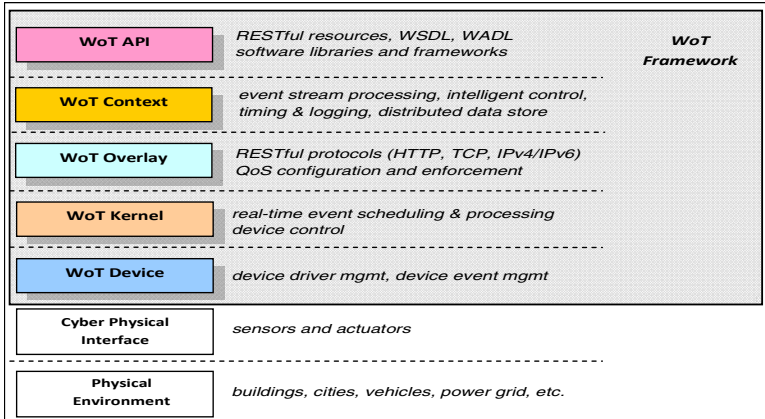
### 3 Brief Overview of Architectural Framework for CPS Systems

We have previously proposed a Web-of-Things (WoT) framework for CPS systems [1] that augments the Internet-of-Things in order to deal with issues such as information-centric protocol, deterministic QoS, context-awareness, etc. We argue that substantial extra work such as our proposed WoT framework is required before IoT can be utilized to address technical challenges in CPS Systems.

The building block of WoT is Representational State Transfer (REST), which is a specific architectural style [4]. It is, in effect, a refinement and constrained version of the architecture of the Web and the HTTP 1.1 protocol [5], which has become the most successful large-scale distributed application that the world has known to date. Proponents of REST style argue that existing RPC (Remote Procedure Call)-based Web services architecture is indeed not “Web-oriented”. Rather, it is merely the “Web” version of RPC, which is more suited to a closed local network, and has serious potential weakness when deployed across the Internet, particularly with regards to scalability, performance, flexibility, and implementability [6]. Structured on the original layered client-server style [4], REST specifically introduces numerous architectural constraints to the existing Web services architecture elements in order to: a) simplify interactions and compositions between service requesters and providers; b) leverage the existing WWW architecture wherever possible.

The WoT framework for CPS is shown in Fig. 1, which consists of five layers – WoT Device, WoT Kernel, WoT Overlay, WoT Context and WoT API. Underneath the WoT framework is the cyber-physical interface (e.g. sensors,

actuators, cameras) that interacts with the surrounding physical environment. The cyber-physical interface is an integral part of the CPS that produces a large amount of data. The proposed WoT framework allows the cyber world to observe, analyze, understand, and control the physical world using these data to perform mission / time-critical tasks.



**Fig. 1.** WoT Framework for CPS

As shown in Figure 1, the proposed WoT based CPS framework consists of five layers:

(1) **WoT Device:** This layer constitute the cyber-physical interface of the system. It is a resource-oriented abstraction that unifies the management of various devices. It states the device semantics in terms of RESTful protocol. (2) **WoT Kernel:** This layer provides low level run-time capability for communication, scheduling, and WoT resources management. It identifies events and allocates the required resources, i.e. network bandwidth, processing power and storage capacity for dealing with a large amount of data from the WoT Device layer. (3) **WoT Overlay:** This layer is an application-driven, network-aware logical abstraction atop the current Internet infrastructure. It will manage volatile network behavior such as latency, data loss, jitter and bandwidth by allowing nodes to select paths with better and more predictable performance. (4) **WoT Context:** This layer provides semantics for events captured by the lower layers of WoT framework. This layer is also responsible for decision making and controlling the behaviour of the CPS applications. (5) **WoT API:** This layer provides abstraction in the form of interfaces that allow developers to interact with the WoT framework.

Based on the WoT framework in Fig. 1, the CPS reference architecture is shown in Fig. 2, which aims to capture both domain requirements and infrastructure requirements at a high level of abstraction. It is expected that CPS applications can be built atop the CPS reference architecture.

More details about the CPS Fabric structure and the CPS node structure are given in Dillon et. al. [1].

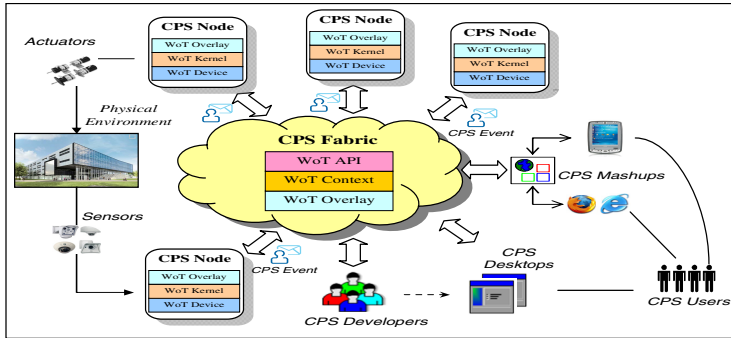


Fig. 2. CPS Reference Architecture

## 4 Semantics for CPS Systems

A key aspect of CPS systems is representing the semantics of events and sensors. Events are one of the key initiators of activities in Cyber-Physical Systems. They provide a natural way to specify components of open systems in terms of interfaces and observable behaviour. Events also can form the basis for specifying coordination and composition of components [8,10]. Due to the wide variety of events ranging from lower level physical signals to higher-level abstract events, CPS applications and users may be interested in certain conditions of interest in the physical world, according to which certain predefined operations are executed by the CPS. Detection of these conditions of interest (events) will lead to the desired predefined operations. As a result, any CPS task can be represented as an “Event-Action” relation [11]. Some of the challenges involved in the representation events in dynamic real-time CPS are [7]:

### A. Abstractions for Sensors and Event Representations

Sensor data requires processing to generate information which can then be used for event identification and representation. Abstraction of sensor instances and event instances are required for representing sensors and events respectively. However, this is a challenging task that requires a framework that can deal with this problem in its entirety. Sensors can be represented as a resource with a unique identifier that allows them to be located through the Internet. In addition, it is important to develop the notion of (1) Sensors classes which represent collections of sensors instances with particular properties. e.g. a temperature sensors monitor and alarm class. This class could consist of several instances. To allow for mobility, this class would have both location and time as two amongst other properties. (2) Event classes which represent collections of event instances with particular properties. e.g. an intrusion event detection class which could consist of several event instances. Event classes too would have location and time (or at least position in a sequence in time) as two amongst other properties.

There will be relationships between these classes, which will allow for representation of generalization/specialization, composition and associations. As mentioned earlier, interoperability is an important challenge to be considered for the seamless integration of different high level applications. Successful implementation of this allows us to meet the following challenges: (1) Management of raw sensor data that is

kept, maintained and exported by disparate sources. (2) Interpreting events associated with particular sensor configurations and outputs. (3) Transforming system level knowledge from distinct sensors into higher-level management applications. (4) Upgrade of existing sensors with new advanced sensors using standardised interfaces by using the same abstract level representation.

### *B. Compositions of Sensors and Events to Address Complex Requirements*

Depending upon specific application requirements, composition of event data from multiple sensors may be required. This is a challenging task that requires a dedicated framework on how information from multiple sensors is composed and correlated for meeting the QoS requirements of the specific application scenario.

A decomposition technique is needed for decomposing complex functionality into lower level resources which can then be met by specific sensors or events. This decomposition requires the specification of items in the aggregation and their dynamic sequence of execution or arrangement. The dynamics in the case of services can be modelled by using workflows specified in a language such as BPEL and in the case of resources by using Mashups which allow easier configuration of these workflows.

The composition of events remains a challenging issue with a focus on sequencing them to produce a composite event.

### *C. Semantics for Compositions of Sensors and Events*

Semantics need to be provided for automatic sensor discovery, selection and composition. Semantics here is the study of the meaning of Things (resources that represent sensors) and events. It represents the definitions of the meaning of elements, as opposed to the rules for encoding or representation. Semantics describe the relationship between the syntactical elements and the model of computation. It is used to define what entities mean with respect to their roles in a system. This includes capabilities, or features that are available within a system.

Two approaches can be used to represent semantics. They are: (1) ontologies and (2) lightweight semantic annotations.

Ontologies are a formal, explicit specifications of a shared semantic conceptualization that are machine understandable and abstract models of consensual knowledge, can be used. Using an ontology, it is possible to define concepts through uniquely identifying their specifications and their dynamic and static properties. Concepts, their details and their interconnections are defined as an ontology specification. Ontology compositions are typically formed from many interconnected ontology artifacts that are constructed in an iteratively layered and hierarchical manner. It is necessary to use a graphical representation of ontologies at a higher level of abstraction to make it easier for the domain experts to capture the semantic richness of the defined ontology and understand and critique each model.

Ontologies can be used to define the following system properties: (1) Information and Communication, refers to the basic ability to gather and exchange information between the parties involved. (2) Integrability, relates to the ability of sensors and devices from different sources to mesh relatively seamlessly and the ability to integrate these sensors to build the final solution in a straightforward way. (3) Coordination, focuses on scheduling and ordering tasks performed by these parties. (4) Awareness and Cooperation, refer to the implicit knowledge of the operation process that is being performed and the state of the system.

For representing the knowledge in a given domain ontologies and for adding semantics to individual resources annotation may suffice. A special challenge here is developing ontologies for events. Lightweight semantic RDF Metadata Annotations are an attractive alternative for providing semantics for CPS systems. The Resource Description Framework (RDF) provides an enhanced representation over XML including: (1) The triples (Resources, Property, Value) or (Subject, Predicate, Object). (2) Defining relationships such as the concept of class and subclass

RDF is extensible, which means that descriptions can be enriched with additional descriptive information. RDF is metadata that can be inserted into XML code or device or vice versa . An example of an Extensible representation of a sensor instance is given in Figure 3 below and associated URL Hierarchy is given in Figure 4..

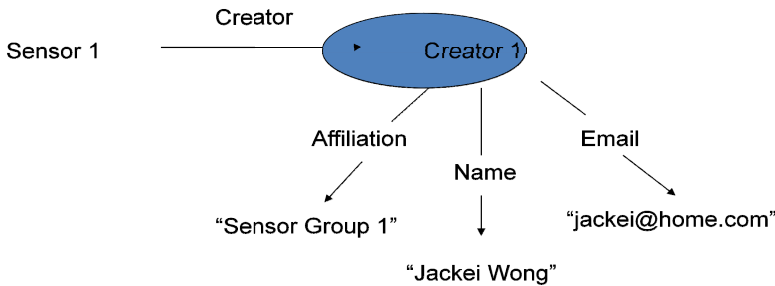


Fig. 3. Extensible RDF representation of a Sensor instance

This RDF statement may also be described in terms of triple (Subject, Predicate, Object). Example:

- Subject: <http://www.webofthings.com/~test>
- Predicate: The element <creator>
- Value: The string “Jackei Wong”

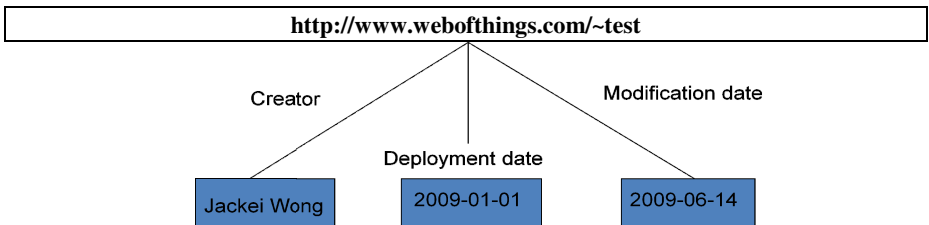


Fig. 4. URL Hierarchy

Another important aspect here would be the use of RDF profiles. A sample RDF Profile is given in Figure 5.

Semantics for CPS systems must be able to represent: Timeliness requirements; Resources descriptions (e.g. hardware or software applications); Quality of service; Schedule of tasks; Other functional issues that must be addressed. Thus one can distinguish four main semantic components namely:Time- provides a detailed definition of



time; Resources- provides a wide range of primitives for resource description including physical resources such as device, CPU or communication devices; as well as resource service such as application or services description; QoS -At this stage we have provided a profile for illustration of QoS concepts for Resource Instances description; Causality-The description of action execution and resource service instance.

<b>Title</b>	<b>Definition</b>
Identifier	An unambiguous reference to the resource within a given context
Description	Purpose of resource and what it does eg. measures xx quantity
Coverage	The extent or scope of the content of the resource
Creator	An entity primarily responsible for creating & deploying the resource
Format	The physical or digital manifestation of the resource
Date	A date of an event in the lifecycle of the resource
Type of resource	The nature or genre of the resource
Relation	A reference to a related resource
Access rights	Information about rights to modify & access the resource

**Fig. 5.** Example of a RDF Profile

#### *D. Event Models for CPS*

Approaches have been proposed in the literature that model events in CPS. They classify events as temporal or spatial [11]. They further define different dimensions such as (a) punctual or interval, (b) single or stream, (c) action or observation, (d) point or field, and (e) causal, discrete, or continuous [10,11].

In a CPS, the events can be further categorised as follows [11]: (1) Physical-events: Physical event models the occurrence of the end-user interest in the physical world and can be any change in attribute, temporal or spatial status of one or more physical objects or physical phenomena. These events are captured through physical observation, which is a snapshot of attribute, temporal, or spatial status of the target physical event. (2) Cyber-Physical events: The physical event captured using sensors collect the sensor event instances from other sensor notes as input observations and generate cyber-physical event instances based on the cyberphysical event conditions. (3) Cyber-events: The top level CPS control unit serves as the highest level of observer in CPS event model. It may combine cyber-physical event instances from other CPS components (sensors) and other control units as input observations to generate the cyber event. Some of these approaches utilise a Spatial-Temporal event models for CPS in 2-dimensions [11] and in another uses 3-dimensions [12]. Another approach provides semantics to the events detected by using first order logic, such as an adaptive discrete event calculus [13].

## 5 Conclusion

In this paper we discussed a semantic framework for Cyber Physical Systems based on the reference architecture previously presented. A key element here is the semantics of Events and Sensors. Both Ontologies and lightweight semantics based on RDF could be used depending on the specific application and domain.

## References

1. Dillon, T.S., Zhuge, H., Wu, C., Singh, J., Chang, E.: Web-of-things framework for cyber-physical systems. *Concurrency and Computation: Practice and Experience* 23(9), 905–923 (2011)
2. President’s Council of Advisors on Science and Technology (PCAST), Leadership under challenge: Information technology r&d in a competitive world (August 2007), <http://www.nitrd.gov/pcast/reports/PCAST-NIT-FINAL.pdf>
3. National Science Foundation, Cyber-physical systems (CPS) workshop series, <http://varma.ece.cmu.edu/Summit/Workshops.html>
4. National Science Foundation, Cyber-physical systems summit report, Missouri, USA (April 24–25, 2008), [http://precise.seas.upenn.edu/events/iccps11/\\_doc/CPS\\_Summit\\_Report.pdf](http://precise.seas.upenn.edu/events/iccps11/_doc/CPS_Summit_Report.pdf)
5. Lee, E.: Cyber physical systems: Design challenges. In: *IEEE Object Oriented Real-Time Distributed Computing*, pp. 363–369 (2008)
6. Lee, E.: Computing needs time. *Communications of the ACM* 52(5), 70–79 (2009)
7. Singh, J., Hussain, O., Chang, E., Dillon, T.S.: Event Handling for Distributed Real-Time Cyber-Physical Systems. In: *IEEE 15th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC)*, China, pp. 23–30 (2012)
8. Dillon, T.S., Talevski, A., Potdar, V., Chang, E.: Web of Things as a Framework for Ubiquitous Intelligence and Computing. In: Zhang, D., Portmann, M., Tan, A.-H., Indulska, J. (eds.) *UIC 2009*. LNCS, vol. 5585, pp. 2–13. Springer, Heidelberg (2009)
9. Dillon, T.: Web-of-things framework for cyber-physical systems. In: *The 6th International Conference on Semantics, Knowledge & Grids (SKG)*, Ningbo, China (2010) (Keynote)
10. Talcott, C.: Cyber-Physical Systems and Events. In: Wirsing, M., Banâtre, J.-P., Hölzl, M., Rauschmayer, A. (eds.) *Soft-Ware Intensive Systems*. LNCS, vol. 5380, pp. 101–115. Springer, Heidelberg (2008)
11. Tan, Y., Vuran, M.C., Goddard, S.: Spatio-temporal event model for cyber-physical systems. In: *29th IEEE International Conference on Distributed Computing Systems Workshops*, pp. 44–50 (2009)
12. Tan, Y., Vuran, M.C., Goddard, S., Yu, Y., Song, M., Ren, S.: A concept lattice-based event model for Cyber-Physical Systems. Presented at the *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems*, Stockholm, Sweden (2010)
13. Yue, K., Wang, L., Ren, S., Mao, X., Li, X.: An Adaptive Discrete Event Model for Cyber-Physical System. In: *Analytic Virtual Integration of Cyber-Physical Systems Workshop*, USA, pp. 9–15 (2010)
14. Yu, X., Cecati, C., Dillon, T., Godoy Simões, M.: Smart Grids: An Industrial Electronics Perspective. In: *IEEE Industrial Electronics Magazine IEM-02-2011* (2011)