Position Paper: Deployment and Configuration – An Important Factor for Realizing High Confidence Cyber Physical Systems

Aniruddha Gokhale
Dept. of EECS, Vanderbilt University, Nashville
a.gokhale@vanderbilt.edu

1 Problem Statement

Many of us have experienced the hazards of driving a vehicle during a thunderstorm or on icy roads. Reckless drivers make the situation worse. Enhancing the safety of the occupants of the vehicles in these as well as in normal conditions is the goal of the planned and emerging cyber physical systems such as intelligent transportation systems (ITS), *e.g.*, Tennessee's SmartWay ITS system [5]. SmartWay is adopting a subset of the 33 different services, including dynamic message signs, traffic visualization and freeway ramp control among others, defined by ITS America and US Department of Transportation. Recent advances [9] in ITS describe intelligent driver support systems.

Given the safety critical nature of cyber physical systems, such as ITS, it is important to focus on developing high confidence systems. Realizing the capabilities of SmartWay ITS requires the different services to be deployed and configured on the different resources (*e.g.,* vehicle on-board processors, sensors, cameras and servers) of the SmartWay system using multiple different technologies including different middleware platforms. One dimension of the challenges in realizing high confidence ITS pertains to how the deployment and configuration mechanisms can assure that the end-to-end quality of service (QoS) requirements of the SmartWay system, *e.g.*, real-time traffic monitoring and communications, are met along with satisfying the constraints on resources, *e.g.*, memory footprint constraints of embedded systems.

Cyber physical systems like SmartWay are increasingly adopting traits from component-based enterprise systems wherein the overall system objectives are realized by orchestrating multiple different services that are assembled and deployed end-to-end on heterogeneous middleware platforms, such as J2EE [8], .NET [3] and CORBA Component Model [4]. Although this service-oriented approach has the potential for the rapid realization of the overall system objectives, a number of challenges related to deployment and configuration of the system must first be overcome before high confidence cyber physical systems like SmartWay can be realized.

1.1 The Deployment and Configuration (D&C) Problem

Deployment and configuration (referred to as D&C hereafter) is an important phase in the developmental lifecycle of distributed systems, particularly those built on the notion of orchestration of distributed services, as in SmartWay. We define deployment as the activity of bootstrapping application components or services and middleware platforms onto the computing and communication resources. Configuration involves tuning the applications and middleware platforms by selecting the right set of tuning options to obtain the desired QoS attributes.

Software engineering processes for D&C in distributed systems have been well researched and the patterns of reuse already documented. For example, the OMG D&C specification [6] provides a clear separation of individual stages of the D&C process. It defines a data model and a runtime model for different entities of the process including the application software components, the resources, and mapping of these components to the resources. A number of D&C tools incorporating these patterns also exist [1, 7, 2] for a variety of component-based middleware platforms.

Although the scientific literature pertaining to the D&C processes is rich, existing tools and documented processes have focused largely on how to deploy and configure application components onto the system resources involving homogeneous middleware platforms, but mostly ignored a dimension of D&C which relates to the deployment and configuration of the heterogeneous middleware platforms that host the application components. This dimension of the D&C space is particularly

important for high confidence cyber physical systems owing to their stringent QoS requirements, such as end-to-end latencies; resource constraints, such as limits on memory footprint; and heterogeneity in the operating environment.

To realize high confidence cyber physical systems, deployment of the middleware requires selecting the necessary features of the middleware and optimizing them for QoS properties. For example, a command and control center of an ITS will require concurrency management solutions. A deployment mechanism will need to select the right concurrency solution within the middleware, *e.g.*, thread-per-request or thread pool. Once the feature is selected, the deployment mechanism must look for performance optimization opportunities within that feature. Middleware configuration involves choosing the right QoS tuning parameters of the selected features of the middleware *e.g.*, size of the thread pool or the stacksize for each thread.

1.2 Technical Gaps in Deployment and Configuration

Addressing this dimension of the D&C problem space is a hard problem primarily for two reasons:

• **Generality of middleware platforms:** Contemporary middleware platforms provide platform-independent execution semantics and reusable services (*e.g.*, concurrency management, connection management, data marshaling, location transparency), which coordinate how application components are composed and interoperate. These middleware solutions are often designed to be general-purpose, highly flexible and very feature rich *i.e.*, they provide rich set of capabilities along with their configurability to support a wide range of application classes in many domains.

Unfortunately, cyber physical systems have stringent QoS requirements and resource constraints, and hence find this feature richness and flexibility to be a source of excessive memory footprint overhead and a lost opportunity to optimize for significant performance gains. There is a need on one hand for cyber physical systems to continue to benefit from the elegant, object-oriented designs and interfaces of middleware for maximum reuse and interoperability. On the other hand, it is necessary for these systems to use only the required features of the middleware and derive maximum performance mileage in response to their QoS needs. Selecting the right set of features, optimizing and tuning them is a challenge. Moreover, it is necessary to ensure that the feature selection and optimizations within the layers of middleware are compatible with each other. *There is however no one single middleware D&C approach that will address the QoS issues and in turn the high confidence issues of cyber physical systems.*

• **Heterogeneity in the problem space:** Heterogeneity is an inherent characteristic of any large-scale cyber physical system. The concept of middleware was developed to overcome traditional sources of heterogeneity, which stemmed from the differences in hardware, operating systems, compilers, databases, and programming languages. However, as evident from systems like SmartWay, a new dimension of heterogeneity manifests itself in the form of diversity in the middleware platforms, particularly due to the differences in the component-based programming and communication abstractions they provide, and their data and runtime models. Middleware D&C mechanisms must address the dimension of heterogeneity when selecting the features and optimizing them since these must be compatible across the collaborating middleware platforms. *There is however no one single middleware D&C approach to resolve the heterogeneity issues that will address the high confidence needs for cyber physical systems.*

1.3 Research Needs

Addressing these challenges requires new scientific approaches to deployment and configuration so that the goals of high confidence cyber physical systems can be met. We do not imply that D&C is the only challenge in realizing high confidence cyber physical systems. Yet it remains to be an unaddressed challenge. This position paper focuses on such a challenge outlining the need for research in this dimension of the problem space. The scientific approaches that need to be investigated will also need associated tools and techniques to verify and validate the correctness of the D&C decisions.

References

[1] A. Akkerman, A. Totok, and V. Karamcheti. Infrastructure for Automatic Dynamic Deployment of J2EE Applications in Distributed Environments. In *3rd International Working Conference on Component Deployment (CD 2005)*, pages 17–32, Grenoble, France, Nov. 2005.

[2] G. Deng, J. Balasubramanian, W. Otte, D. C. Schmidt, and A. Gokhale. DAnCE: A QoS-enabled Component Deployment and Configuration Engine. In *Proc. of the 3rd Working Conf. on Component Deployment (CD 2005)*, pages 67–82, Grenoble, France, Nov. 2005.

[3] Microsoft. .NET Web Services Platform. `www.microsoft.com/net`.

[4] Object Management Group. *CORBA Components v4.0*, OMG Document formal/2006-04-01 edition, Apr. 2006.
[5] T. D. of Transportation (TDOT). SmartWay Strategic Plan, Dec 2006.
[6] OMG. *Deployment and Configuration of Component-based Distributed Applications, v4.0*, Document formal/2006-04-02 edition, Apr. 2006.
[7] S. Paal, R. Kammtiller, and B. Freisleben. Crosslets: Self-managing Application Deployment in Cross-Platform Operating Environment. In *3rd International Working Conference on Component Deployment (CD 2005)*, pages 52–66, Grenoble, France, Nov. 2005.
[8] Sun Microsystems. Java^{TM} 2 Platform Enterprise Edition. java.sun.com/j2ee/index.html, 2001.
[9] M. M. Trivedi and S. Y. Cheng. Holistic Sensing and Active Displays for Intelligent Driver Support Systems. *IEEE Computer*, 40(5):60–68, 2007.