National Workshop on Beyond SCADA: Networked Embedded Control for Cyber-Physical Systems (NEC4CPS): Research Strategies and Roadmap

Bruce Krogh, Marija Ilic and Shankar Sastry (PI)

Final Report

November 27, 2007

CONTRIBUTORS

Clas Jacobson, United Technologies Corporation
Timothy Johnson, General Electric Company
Al Mok, University of Texas at Austin
Tariq Samad, Honeywell International Inc.

ANNUAL REPORT

This annual report is a draft version of the final report to be published by the National Coordination Office (NCO) of the NITRD. The final version of this report will also be the final report for the NSF grant to support the two workshops:

1. National Planning Workshop, March 16, 17, 2006 and
2. Final National Workshop held November 8,9, 2006.

The details of the participants, program, and the presentations at the workshop and discussions on a Wiki site are available at http://truststc.org/scada (See also Appendix 1 and 2 of this report for this information for the second of the workshops).

ABOUT THIS REPORT

This report is the third in a series of prospective technical studies ("report" and "study" will be used interchangeably throughout this document) conducted by the High Confidence Software and Systems (HCSS) Coordinating Group (CG), one of seven Program Component Areas (PCAs) of the Networking and Information Technology Research and Development (NITRD) Program, a part of the National Science and Technology Council (NSTC), Executive Office of the President. Sponsor agencies for the workshop include NIST, NSA, and NSF, along with the National Coordination Office (NCO) for NITRD. The NSTC is the primary mechanism in which the President coordinates science and technology across the Federal government domain.

The report's objective is to focus national attention on the importance of identifying the R&D required for designing, deploying, and operating the next generation of high confidence supervisory data acquisition and control (SCADA) systems and distributed control systems, broadly referred to as networked embedded control for cyber-physical systems (NEC4CPS). These are the technologies that monitor and control our Nation's critical infrastructures and production processes. The other domain-specific reports address the R&D required for developing the next generation of high confidence aviation systems and high confidence medical devices. The final report in this series will focus on the R&D required for building a more sound and assured, real-time technology base to mitigate risks associated with the rapidly increasing complexity of IT-enabled devices and systems. The HCSS CG expects this series of reports will help shape national research investments towards this end.

Material in this report was compiled from the intellectually diverse presentations and discussions, breakout session reports, and white papers provided at the NEC4CPS Workshop held November 8-9, 2006 in Pittsburgh, Pennsylvania. The technical sections of this study, in large part, were developed by a group of leading university and industry researchers from the areas networking, control, and relevant application domains who voluntarily served on the NEC4CPS Workshop's Program Committee (PC). (See Appendix 1.)

This report, a tangible outcome of the workshop, provides a full account of the NEC4CPS Workshop proceedings as well as the technical findings, research challenges, and the fundamental elements for a research roadmap to determine what, when, and how priorities should be addressed over identified time frames. Part 1 provides a synoptic view of the major technical findings including the research needs, new research directions, and promising technical approaches for building the next generation of high confidence network-enabled control technologies that are crucial to innovation and to the operation of our Nation's critical infrastructures. Part 2 discusses in detail the major technical issues identified requiring innovative research and development strategies. Part 3 summarizes the views of invited academic and industry speakers to the workshop, and explains Government agency (i.e., regulatory, practitioner, and IT research) broad scientific interests in the area of network-enabled control R&D and their motivation for the workshop and report. Several appendices are attached including the HCSS CG agencies, workshop program committee, report contributors, workshop participants, workshop agenda, definitions and acronyms, and acknowledgments.

Part 2: Technical Perspectives and Analyses

The following sections discuss in depth the state of the technology and the directions for research in NEC4CPS identified by the working groups at the workshop.

2.1 Emerging Capabilities

2.1.1 Introduction

We begin by identifying application needs and requirements for networked embedded control for cyber-physical systems, with particular attention to SCADA and DCS systems. We eschew discussion of specific technologies and solution approaches—these are covered in later sections. Instead, we identify and elaborate on fundamental capability improvements that networked embedded systems must exhibit so that they can be better and more widely deployed to benefit the nation's safety, security, and economic progress.

We envision a class of SCADA, DCS, and other automation and control systems that can be trusted by their owners and users in safety- and mission-critical domains. This trust must encompass a variety of aspects: from design to implementation to phaseout, from operation under nominal to abnormal conditions, from component behaviors to system properties to human-system interactions. We highlight below some elements of the vision (expressed in terms of capabilities) we hold for NEC4CPS:

- **NEC4CPS will be adaptive and agile**. They will have the capability to self-heal and self-correct in the face of potential, imminent, or occurred failures.
- **NEC4CPS will be heterogeneous and dynamically reconfigurable.** They will be composable and scalable during operation, offering in-field extensibility with components and subsystems from different suppliers without downtime or shutdown.
- **NEC4CPS will encompass physical, cyber, and enterprise systems.** They will allow information to be accessed and actions to be taken across today's digital divides.
- **NEC4CPS will provide cradle-to-grave life-cycle management.** They will allow end-of-life predictions on a continuing basis and will continue to provide desired levels of performance under aging and degradation.
- **NEC4CPS will provide situational awareness** to human users, enabling them to make the right decisions in high-stress, safety-critical circumstances.
- **NEC4CPS must be human-in-the-loop-aware**, sensitive to the limitations of human users and to the possibility of insider attack.
- **NEC4CPS may be autonomous.** In some cases an NEC4CPS may operate without any human supervision for its entire useful life.
- **And yet ... NEC4CPS must be secure, reliable, fault-tolerant, robust.** They cannot compromise, beyond acceptable bounds, safety and security for performance.

The following sections discuss where we are with respect to these capabilities and the research challenges we face to realize them fully.

2.1.2 State of the Technology

Simultaneously with painting the vision for NEC4CPS, we think it is essential to recognize both the achievements and the shortcomings of cyber-physical systems today. The quality of life that we have attained in modern societies is, to a significant extent, a testament to the success we have had in developing and deploying complex engineering systems. By and large, the nation's citizens live safely and securely. Our complex systems are the products of complex processes wherein we continually balance performance needs with reliability, safety, and security—and systematically bias design and operation toward the latter. However, the complexity of the systems is such that they can never be made failure proof, and in fact the increasing information technology and networking content in our networked embedded systems is creating new problems (while offering new and better solutions).

We note three examples of NEC4CPS to help provide the application context for emerging capabilities. Consider first the process industries, and more specifically petroleum processing. A refinery automation system (composed of one or more distributed control systems) is a large-scale NEC4CPS that may contain 10,000+ sensors, 10,000+ actuators, and 10M+ lines of software, and may be used by 100s of onsite and offsite personnel. The potential for catastrophe is huge—apart from the risk to the plant and its staff, many U.S. chemical and petrochemical plants and refineries are located sufficiently near large urban areas that hundreds of thousands of people could be injured or killed. Accidents have occurred, with casualties, in this country, and we need only recall the Bhopal tragedy of December 2004 to recognize the extent of loss that is possible.

Our second example is at the other end of the physical scale. Biomedical devices are on a steep growth curve and are also steadily increasing in software content and functional complexity. Initial pacemakers, for example, were "worn" by their users and were hardwired to provide fixed periodic stimulation. Today's devices are miniaturized implants that can sense the current state of the heart, ascertain whether or not stimulation is needed, and deliver a stimulus of the right intensity at the right time. The advance in biomedical devices is a clear net benefit to society, but increasing complexity has led to new failure loci and modes.

Finally, we present an anecdote that highlights the increasing interdependency of subsystems today. A large European manufacturer of luxury automobiles had incorporated what it thought were foolproof antitheft mechanisms. As a final check it brought in expert car thieves and asked them to attempt to defeat the mechanisms. The car thieves jumped on the roof of the car, rocking it, and the car doors unlocked. The thieves knew that the manufacturer's cars were designed to unlock in case of a rollover and that roll accelerometers were used to detect rollovers!

2.1.3 What we can't do well

The diversity of NEC4CPS extends to capability requirements. Systems differ in scale, complexity, criticality, and other respects that substantially influence the levels of safety and security they must be designed for. In this section we further discuss several of the "visionary" capabilities noted above, with the objective of helping identify topics that could be elements of a research agenda and roadmap.

Integration of heterogeneous systems and services. Interoperability, plug-and-play, composability: these are key requirements for many cyber-physical systems. We foresee a future where different suppliers can provide components and subsystems that can readily be integrated into NEC4CPS. These components and subsystems can include hardware, software, networks, information, requirements, etc. The issue of integration of new systems and services with legacy systems is of particular importance for SCADA and DCS systems, given their extensive deployment—in many cases with products that are decades old.

Particularly from a high-confidence perspective, there is a need for integration mechanisms that incorporate not only physical form factors, low-level communication protocols, and the like, but also encompass requirements, capabilities, semantics, and knowledge bases. We foresee composable elements of networked embedded systems that are reliability-aware and, when "plugged in" to the larger system, can be integrated with similar information from other subsystems to help assess the overall level of confidence at the system level.

SCADA and DCS systems interconnect the cyber and physical worlds. Economic and performance considerations are driving larger-scale integrations, often creating security concerns. Industry seeks connectivity from enterprise applications to sensors and actuators. The research community must ensure that such connectivity is sufficiently secure.

At the level of services integration, interest in service-oriented architectures has now migrated from offline applications (e.g., online retailing) to real-time systems. But this technology has so far been developed with relatively little regard for the requirements of high-confidence safety-critical systems—a shortcoming that needs to be addressed.

Finally, the topic of standards is covered in more detail elsewhere in this report, but we would like to highlight its importance in the current context. Standards (and metastandards) are essential if networked embedded systems are to be cost-effectively constructed from disparate components developed by different vendors and enhanced over time by incorporating new functionality.

Life-cycle Management. The systems of interest to us cannot be "high confidence" at selected points in time; we must be assured of their safety and reliability throughout their deployed life. There are several corollaries to this assertion. For example, networked embedded systems will not be static over the duration of their use but will undergo changes. These changes include degradation caused by normal wear-and-tear as well as unexpected equipment failure, system enhancements and component replacement, and even integration of previously installed systems as “legacy” components in newer products (as noted earlier).

Software-intensive systems are of particular interest in this context. Unlike purely mechanical systems, software upgrades are both easier to perform and it is harder to predict the full consequences of the changes. Mechanisms to improve the confidence in an installed system create their own uncertainties. Software security issues such as viruses result in the need to load “patches” frequently, but unless care is exercised these patches can have unintended consequences. NEC4CPS such as distributed control systems for oil refineries and chemical plants today make extensive use of commercial software systems such as Microsoft Windows. At the same time, however, the Windows PCs in a DCS include many additional applications as well as customized mechanisms for integrating plant hardware and other equipment. Some DCS manufacturers provide a “qualification” service to their customers—before a patch is downloaded extensive tests over days and weeks are done by the vendor to ensure that system behavior will not be affected in unintended ways.

High confidence throughout a system’s life cycle also means that maintenance and related aspects must be proactive activities. More data must be collected, system performance must be monitored continually, and condition-based and predictive maintenance must be employed. As with other capabilities, modeling will be a key element of any solution, and in this case the models will themselves need to be adapted and updated over time.

With models and other understanding of failure and degradation phenomena, prospects for life-cycle optimization can be pursued and ultimately realized. Not only should we be able to predict problems before they fully manifest themselves and to fix them before system reliability is compromised, but system operation should be managed to maximize the useful and safe life of the system (as appropriate). For example, an early indication of a problem with a component (such as a sensor or actuator) could result in a changed control strategy that reduces the dependence on that component. Analogously, knowing that a spare part may not be available for some period of time could lead to relatively conservative operation until then.

As implied by these remarks, risk assessment and risk management are crucial to life-cycle management. Investment, reliability, and risk levels are products of design tradeoffs, especially in light of (unavoidable) resource constraints,. Safety, security, and reliability are not all-or-nothing properties.

Autonomous and Semiautonomous Systems. The trend toward increasingly autonomous systems—engineering systems that rely on fewer, and even no, human operators—is being driven by both economic and safety considerations. Virtually complete autonomous operation is a goal for many safety-critical applications and it is already the reality in several—from implanted biomedical devices to electricity substations to wastewater treatment facilities. Even some chemical plants now operate without on-site personnel, relying on communication networks for remote oversight. In many other cases, where the full replacement of human operators with automation is not envisioned within planning horizons, workforce reductions are anticipated.

Autonomy thus represents both a challenge and an opportunity for NEC4CPS. The opportunity consists in the ability to keep our citizens out of harm's way (e.g., guided munitions, unmanned vehicles and spacecraft, operator-less automation in hazardous environments) and also to develop sophisticated automation that can sense and react faster than a human possibly could and without the possibilities of fatigue and operator error. On the other hand, the desire to reduce the number of human operators for cost reduction has obvious adverse implications for safety.

Regardless of the motivation, though, there is clarity on requirements. We need to develop more reliable technology that can be trusted even in the absence of constant or frequent human supervision as well as technology that can assist increasingly overburdened operators to manage complex systems without compromising safety.

Autonomous systems is one of the "hot topics" in a number of engineering research communities already, but we believe that insufficient attention has been paid to its essential linkage with high confidence. We would argue that autonomy implies high confidence: the absence of a human in the loop can only be tolerated if the producers and users have sufficient trust in the system. For SCADA and DCS systems, this trust must extend beyond an expectation of safety, for example in the presence of equipment failure caused by normal wear and tear, to an expectation that lives or property or the environment will not be harmed under foreseeable deliberate attack scenarios. Both cyber and physical security challenges must be addressed, individually and in coordination.

"Semi-autonomous" systems pose their own issues, and technology developments can help with these too. Humans are in the loop often for safety, but as noted above they are subject to fatigue and to other causes of mistakes. Effective monitors for human-in-the-loop systems such as DCSs would help improve their operational reliability and safety.

Human operators also increase the possibility of insider attack, with a potential for catastrophic impact that scales with system complexity and distribution. Today's DCSs may evolve to geographically (not just logically) distributed automation systems for multiple plants, bringing regional or even worldwide safety-critical systems within the boundaries of one networked system. The connectivity would not be purely cyber; supply-chain and physical dependencies (consider, respectively, fuel supply for a coal-fired power plant and the electrical transmission of its output) would create inter-networked systems with huge opportunities for efficiency and performance ... and huge challenges for safety and security. Secure functioning of interconnected, coordinated, and widely distributed industrial processes, with malicious insiders interspersed among trusted operators (who are nonetheless subject to human failings), is a long-term R&D milestone for human-centric-yet-highly-automated complex systems.

2.1.4 R&D Challenges

The discussion above is not specific to particular domains. Here we discuss research challenges and IT hard problems in three important domains for NEC4CPS: manufacturing, power systems, and intelligent transportation systems. Each of the subsections below has been prepared by an expert in the associated field.

Manufacturing. In looking at the challenges of designing, implementing and maintaining SCADA systems in today's manufacturing facilities, using automotive and semiconductor manufacturing as two examples, the problem can broken down by examining it from four perspectives: data collection, decision making as part of the control loop, modeling to support controller design and verification, and barriers to widespread advancement of SCADA R&D.

From the data collection perspective there is a need for solutions to problems of data consolidation and coordination across the facility and enterprise. Data must be collected from disparate sources and the appropriate knowledge extracted to provide the necessary information to drive the control system. Challenges of time and domain synchronization, common languages for data reporting, and specifying/maintaining data quality fall into this category.

At the decision level, there is a need for knowledge translation and integration from heterogeneous systems, and this knowledge needs to be usable and tractable over time. For example, a large portion of the decision process in future SCADA systems in this domain will rely on modeling of components and systems. Information from these models will come from multiple data sources, empirical knowledge, heuristics, etc. This information must be integrated in a configurable yet standard way. In other words there is a need to integrate software and hardware capabilities together to achieve system goals and to support incremental reconfiguration. Challenges of performance metrics, integration and certification of heterogeneous components, and reconfigurable capability-based control system design fall into this category.

From the modeling perspective there is a need for interfacing and capability sharing among and between real and cyber components. This will facilitate the design and verification of SCADA systems against physical models, interchangeability of cyber and real system components to support pre-verification and incremental deployment, and rapid prototyping of SCADA systems and system components. Issues of capabilities refinement, model abstraction (for both physical component and SCADA models), interface standards, and certification would have to be addressed. Additionally, with respect to control models, there is an increasing need for adaptive model/condition-based control and diagnostics that provide capabilities when the complete control/diagnostics environment is not known, i.e., where there are only a limited number of parameters available to understand the space. Further these systems need to be able to adapt during run-time as new information is ascertained about the environment. Challenges here include providing model-based adaptive control and diagnostics as well as continuous improvement capabilities.

From the perspective of barriers to wide-spread advancement of SCADA R&D, we need to make this technology practically accessible to everyone. Specifically we need tools that will allow for rapid prototyping. We also need to support development and verification in a virtual environment and the standards and certification practices that will facilitate the interoperability and interchangeability necessary to make this practical. Further, we need to define, in a modular way, component capabilities with which SCADA systems will interact, and standards and certification procedures that will allow for the development and pre-verification of these component capabilities so that they may be easily integrated in a heterogeneous SCADA system. Finally, we need to provide solutions that will allow us to have confidence in “machine-to-machine” peer-to-peer communications. This confidence will facilitate capabilities such as rapid prototyping, control system verification, and incremental deployment in a heterogeneous environment.

Power Systems. The power grid is often presented as the poster child for complex networked embedded systems, and for good reason. There are several new capabilities that we envision for power systems that will make step changes in the degree of confidence with which they can be operated.

Power flow control mechanisms remain the holy grail for power systems. These mechanisms can be thought of as routers for power. The only things we have available right now are very expensive power electronic devices such as AC/DC converters. Low-cost technology that can flexibly route power flow would have a revolutionary impact on electricity security.

Large-scale simulation that includes the power grid already exists, but tomorrow’s need is for simulation testbeds that encompass but go beyond the grid. Grid operation depends also on the communication and computing layers involved in monitoring and control, plus the fuel supply chain. All of these elements must be modeled within broadly accessible testbeds.

Operators today are often overwhelmed by massive numbers of alarms when the system is in an abnormal condition. There is a need to evaluate, prioritize and respond to these alarms in better ways than is done today. Wide-area situational awareness displays and analysis tools are specific needs.

Another key capability need is global protection and preventive strategies. Almost all of our protection devices operate on local measurements and local action. This is because there is very little time available to make a decision on what to do when something abnormal happens. Plus, there is the advantage that the operator can "blame the automatic controls" when things go bad. There is a need for diagnostic software that provides operators with suggested courses of action.

The robustness/performance tradeoff exists in every complex system. For power systems, a tool to quantify the margin to the boundary of acceptable operation would help us operate the grid with higher performance and capacity without compromising security.

The problem is challenging since the boundary of acceptable operation depends on the path to the boundary.

From the automobile to intelligent transportation systems. New issues are emerging that are related to information technology insertions in automobiles and in traffic infrastructures. Initiatives in automobile safety have resulted in a number of innovations over the last decade or two. Most recently, electronic stability control has become widely available—by 2011 all new vehicles will likely have a form of stability control as a standard feature in the U.S. This system is a combination of powertrain, brake application, and possibly steering. Some modules and functions in such a networked system may not be exercised until an emergency situation occurs, thereby increasing the criticality of fault tolerance—for example, the system must be tolerant of failures while in the act of preventing a rollover event. Similar issues and concerns arise with other current and projected safety systems, such as traction control, lane departure warning, adaptive cruise control, and platooning.

The USDOT VII initiative proposes the use of short-range radio communications to allow vehicles to communicate with each other. Applications of interest include safety features such as collision avoidance and vehicle alerts to facilitate first-responder traffic. Radio communication of live traffic and navigation data is also envisioned, with an objective of real-time traffic management. Initially this information would be displayed for the driver to act on. However if the driver chooses to ignore the advice the traffic system as a whole must compensate.

Different levels of security and interaction must be considered for intelligent transportation systems. Some information could improve public safety if widely available whereas access to other information will need to be tightly controlled. To minimize the likelihood of spoofing, authentication mechanisms will be needed for first-responder vehicles and traffic advisories. Increasing use of electronic and wireless toll collection, including context-sensitive tolls that could be based on time of day, traffic conditions, and even pollution levels, also have security implications.

2.1.5 Research Strategy and Roadmap

	3-5 years	**6-10 years**	**11-15 years**
Agility/reconfigurability	Rapid, trusted integration of new components and subsystems in legacy DCSs	Real-time reconfiguration of control strategies to threat-level changes	Automatic restructuring of physical system and its automation under severe failure conditions
Autonomy	Integration of physical security sense and respond in remote SCADA systems	Automated safety-sensitive performance rollback in complex critical systems	Trusted autonomy under coordinated cyber and physical attack
Enterprise and services integration	Cross-industry data, communication, and knowledge base specifications for NEC4CPS	Secure enterprise-to-actuator connectivity	Real-time service-oriented architecture for critical applications

Lifecycle management	Cradle-to-grave failure models for control system equipment	Adaptive, closed-loop risk management algorithms and tools	Safety-assured lifecycle advisory system for critical infrastructures
Human-centricity	Monitors for human-in-the-loop automation systems	Insider attack alerting for networked control systems	Secure automation with multiple "friend-or-foe" operators
Education and training	Textbook and course on complex engineering systems for nontechnical majors	Certification policies for designers and users of NEC4CPS	
R&D infrastructure	• Distributed, open SCADA laboratory • Commercial-grade DCS as a national research test bed	• High-fidelity simulation testbed for networked, heterogeneous systems • Multimodeling infrastructure for heterogeneous, large-scale systems	• National centers of excellence for networked embedded systems with academic, industry, and government partners
Technology transfer	High-confidence SCADA system prototypes for electricity substations and oil and gas pipelines	High-fidelity-simulation-based situation awareness tools for power system operators and analysts	Operator monitoring and insider attack detect-and-respond system for commercial DCSs

2.2 Security, Safety, and Certification

2.2.1 Introduction

Networked embedded systems, of which SCADA (Supervisory Control and Data Acquisition, used predominantly in the power industry) and DCS (Distributed Control Systems, used in the process control industry) are common examples, provide the communication between centralized "control room" operations and field devices such as control system sensors and actuators, as well as performing many other associated functions (sensor signal conditioning, signal routing and distribution, data format translation) in many key infrastructure segments. Legacy equipment in the field, having embedded software, is extremely widespread, with current asset value in the US of well over $100B in process and utility segments alone. Aside from its cost, this equipment also performs on-line 7x24 critical operational functions and in many cases does not have "hot backup" capability. *It is the real-time, critical, and embedded nature of this equipment that distinguishes its Information Technology, and in particular, security, requirements from other IT applications and procedures.* The annual replacement rate of such equipment (5-10%) is not high enough to rapidly achieve security through yearly upgrades (since security is a system-level property derived from both old and new equipment), but also, there are *no currently suitable concepts or methods for designing and implementing secure embedded (hardware or software) components*, meaning that interim solution methods and processes must be employed to facilitate a gradual transition to a more secure environment over the next two decades. Finally, many new and emerging technologies such as wireless and web-based interfaces, and wireless sensing networks, introduce new and potentially severe security threats that also demand immediate action, since the widespread use of these technologies cannot be delayed merely by regulation, due to their potentially significant performance improvements or cost savings. The key problems identified by this working group expand on these issues.

Secure technologies are widely perceived as being expensive to purchase and to involve costly operating procedure changes, and the compensating benefits, though clear at a national level, are not clear to individual businesses. One workshop contributor has accumulated a list of approximately 80 incidents (some not publicly available) of utility system security threats in the US. *Added benefits of secure technologies must be traded-off against existing performance metrics such as reliability, throughput and latency, and an acceptable level of cost and risk must be determined*. There is currently also a lack of commonly accepted security goals, metrics, and standards that is a barrier to identifying specific research objectives for responsive solutions, and should be a focus of early education and training in this area.

The following areas were considered to be of particular relevance for future research:

1. *IT security concepts and practices suitable for networked embedded systems are needed*

The extension of IT security concepts and procedures from conventional to embedded systems is essential for assuring security for public infrastructures, such as utilities, water, process controls, and transportation. Embedded systems have unique needs, such as maintaining closed loop operation during upgrades, the need for strategies for combined hardware/software upgrades, and support for continuing upgrades due to the long field life of associated assets (sensors, actuators, etc.) relative to software and security upgrade rates. At the same time, the widespread current practice of leaving old hardware in the field with no maintenance, assurance, or authorization mechanisms is also unacceptable, and it is recognized that significant improvements in IT management methods are needed. The number of individuals who currently understand both IT management (and particularly security management) and embedded control system operation is very small, but standardized methods and procedures for these systems need to be developed and widely disseminated. Since many embedded systems are designed to operate for long periods without human intervention, there has been under-attention to the human aspects of embedded system security, such as user authentication.

2. *Performance measures for security and their relationships to existing performance measures for embedded systems throughout the life cycle are needed.*

Embedded systems, and particularly control systems, must be carefully designed in advance of deployment, due to the often-serious consequences of software or hardware failures. Existing performance measures, for which metrics exist, are reliability, stability, throughput, and latency. *Security vulnerability risks and measures need to be identified and integrated into the design frameworks commonly used for embedded systems, so that system security performance can be characterized and traded-off versus conventional design measures.* In some regards, extensions of existing safety and reliability design tools such as the failure modes and effects analyses (FMEA), safety use cases, and the theory of uncertain systems, may be generalized to achieve these objectives, but in other cases, fundamentally new design measures need to be considered, such as "vulnerability" of a design to mis-use or mistakes (instead of intended operation), authentication of command values, and physical system integrity through decentralization of functions. Military system design concepts may be useful here.

3. *Methods for cost-effective evolution from legacy equipment to secure, networked embedded systems.*

Since threats, unlike normal operating conditions, do not remain static over the equipment life cycle, means for continually upgrading fielded equipment to maintain security to all known threats (denial of service, viruses, etc.), need to be invented. Accommodating threats usually requires larger changes than are covered by traditional robust or adaptive control design, but may also require new built-in capabilities for upgrade or alternate operating modes that can be used when threats are detected; existing design methods should be fundamentally augmented to accommodate these needs.

The huge amount of legacy equipment in a wide variety of operational facilities is considered to be such a sufficiently large barrier to adoption of new technology that it warrants independent research investment. Existing systems tend to be "only as secure as their weakest link", and hence new enhancements in upgraded subsystems may not result in significant system-level security improvements. A strategy and technical approach for upgrades of legacy equipment needs to be developed. "Bump on the wire" technology as proposed for SCADA data link security is an example of this approach, but evolutionary methods for step-by-step improvement of system level security through a sequence of ordered upgrades of specific device types, perhaps starting at the lowest levels, is needed. This is a legitimate systems research area in itself, and is compatible with a longer-term goal of life cycle IT security management.

4. *Metrics, Standards, and Certification*

High confidence has been characterized as "the human reaction to the difference between expected and actual system performance." The real-time aspect of confidence is important, and may depend on the relatively intangible "expected performance". Unlike conventional real-time embedded performance metrics (latency, throughput, reliability, or safety), security has been measured in terms of "checklist compliance", without and fundamental basis for checklist items. New security metrics, suitable for design tradeoff with other performance metrics, need to be developed. These might be related to perimeter security, levels of defense against attack, or grades of attack sophistication, for instance. Metrics need to gain wider understanding and common acceptance. Embedded safety systems have been subjected to *safety integrity levels* (SIL) levels based on criticality, and perhaps could be extended. Both performance and process-based standards are in use for safety assurance, and similar procedures are probably needed to assure security. Emerging technologies such as wireless communication and distributed sensing will require the development and extension of conventional metrics and standards. Some key objectives of standards are to assure interoperability among devices of the same security level, and to assure that systems built from subsystems at a given security level will also exhibit the same security level. Certification is a procedure for ensuring that a given subsystem or system meets a certain standard. At the system level, due to the wide variety of legacy equipment in the field, effective procedures are needed for assuring that both security processes and equipment are certifiable. The approach to certification should have standard, quantified objectives, although actual certification test cases will probably need to be specific to each installation. Unlike existing certification approaches, security breaches can induce non-intended modes of operation that devolve over time, or may exhibit intelligent and adaptive modes that are non-deterministic so that conventional regression testing may not work well.

2.2.2 State of the Technology

Although they remain a topic of active research, embedded systems have become ubiquitous. Highly safe, reliable, and rapid-response open and closed loop systems can be designed and are in wide use. These perform well in the vast majority of cases, when a passive adversary ("nature") gives rise to non-malicious faults. Military systems have been designed for increased robustness and (although vastly more complex and expensive than commercial systems) have functioned effectively in field situation where malicious threats are present. Interim methods have also been identified for improving the security by "patching" existing systems, by access control, defense-in-depth strategies, and by authentication, but these are not in wide use yet.

2.2.3 What we can't do well

There are no commonly accepted definitions of security today, and no methods for "building in" security to a new embedded system design, particularly software (cyber-) security and combined hardware-software (cyber-physical) security. There is a disjunction between current concepts of "IT Security" (for a small number of standardized platforms) and "embedded cyber-security", although in most cases these systems interoperate. Furthermore, system design methods for embedded system do not very clearly represent hardware aspects of software systems or human computer interactive aspects of displays. Without even a representation of important security effects, good designs cannot be achieved. Metrics, standards, and certification for security exist only in very restricted circumstances (e.g., nuclear power plants or certain military systems), and in those cases, the costs are enormous.

"Victory" for secure embedded systems is not even close! It cannot be achieved by simply patching legacy systems. It requires the ability to design secure systems and performance metrics. It requires fundamentally new entities (threat models) in the design process, and consideration of new (damaged) modes of operation. It requires design approaches that incorporate better IT security and maintenance practices, and this has barely been attempted yet. High confidence secure systems cannot be achieved without their embedded components being secured; these systems are the key boundaries between physical security and cyber-security. IT procedures for embedded systems will require continual upgrade and maintenance processes that are at least as rigorous as those of desktop and mainframe systems.

2.2.4 R&D Challenges

Several research areas have significant potential for resolving the barrier problems associated with achieving secure, embedded high confidence systems. These have been grouped in the areas of embedded security, evolve-ability methods for legacy systems, metrics, and standards and certification. Each of these is briefly discussed.

Embedded security

Secure operation requires a "defense in depth" strategy, with each level having the ability to detect intrusion, determine a remediation strategy, and to see through its execution. Although fault-tolerant design methods have been developed to accommodate selected "natural" faults, secure-tolerant design methods have yet to be invented. These methods may require systems that are not only self-aware, but also aware of human operators and able to differentiate legitimate from illegitimate operation (intrusion). These objectives need to be achieved concurrently with other ongoing embedded system functions.

Evolve-ability from legacy systems

In the near term, there is an urgent need to define IT procedures suitable for continual improvement from the existing non-secure legacy embedded systems toward fully secure networked embedded systems with embedded security. A substantial toolkit for subsystem upgrade, and an ordered, general-purpose procedure for moving from a non-secure to a secure embedded operating environment should be determined. These advances are not simply applications of known methods, but are in themselves a topic for active research. IT procedures and practices should be internally supported by operating systems and at the lowest application levels of embedded systems. Such methods need to take into account hardware/software tradeoffs (e.g., FPGA's for embedded code), hot swapping and other availability-preserving upgrade methods, on-line upgrade and verification methods, and integrated remote diagnostics). When this is finally achieved (perhaps in 30 years, based on innovations realized as new products within 5-10 years), the vision is that embedded systems will be capable of continual upgrade and renewal in a manner similar to what is being achieved for networks and desktop systems under routine IT management today. Security threats will not stand still, and hence "evolve-ability" will necessarily be a feature of future secure systems.

Security metrics

A security metric is not simply a qualitative concept, but a well-defined quantitative and measurable variable (or variables) that is monotonically related to "perceived" cyber-physical security, i.e., high confidence. To be valuable in practice, such metrics must also enjoy wide acceptance among designers and embedded system users. Although current quantitative design metrics such as reliability and availability are accepted metrics that touch the edges of cyber-physical security, they do not in themselves directly consider or assure security. High confidence implies that a user has a high level of belief that a system will operate "as expected" through a wide variety of the most likely cyber and physical threats. Research is needed to accurately characterize and/or classify such threats in an "open" environment, and then to identify suitable metrics for each dominant threat type. Cyber-secure design methods, as previously indicated, attempt to maximize the values of security metrics (normally, in a trade-off with other metrics, and subject to dollar cost limitations): So, to be viable, a security metric also should be easy to integrate with current design methods. Present security metrics are frequently characterized by "check-lists" (e.g., number of items "checked") or "tests" and may bear little rigorous relationship to real security in the field. Cyber-security metrics need to be invented, verified, and associated with new design methodologies. They need to be widely publicized and accepted by experts in this field, and supported by demonstration test beds that exhibit high confidence operation. All of these stages have been carried out for traditional IT systems, but none of them have been carried out for embedded or closed loop systems.

Standards and Certification

Although standards and certification often coalesce as a technology becomes mature, some fields (e.g., communications) benefit from the early definition and subsequent evolution of widely accepted standards. Cyber-physical security appears to be one such field, since few customers are likely to purchase new systems (particularly in a market dominated by legacy equipment) until they have some assurance of its industry wide acceptance and compatibility: a *single* new threat type can make an *entire* security investment almost worthless. No buyer is willing to assume this much risk. A user community consensus, captured in a standard, and terms of reference (including metrics) is required. The fact that old and new technologies must be combined in many embedded infrastructure applications, and that interfaces to subsystems typically have extra (trust) vulnerability, implies *that both process and performance-based standards will be required*. When two systems with the same security level are combined, the resulting system should be assured to have at least the same security level, an *interoperability* requirement; this is somewhat analogous to the means used to classify and assure "safety integrity levels" (SIL's). While process-based standards can generally be verified via audit processes, performance-based standards require a *certification* procedure. As threats evolve rapidly, such procedures may need to be tailored to be *extensible and (easily) repeatable*, much more than with most present embedded equipment. In addition, the advent of new hardware and software technologies, such as wireless devices and self-aware or evolutionary algorithms, require that certification methods be developed for non-deterministic systems (i.e., systems that may produce different results for the same set of input data), e.g., inputs in one *set* produce outputs always within another *set*, even though the response to a particular input may not be precisely predictable. In summary, standards and certification should *in themselves* constitute a topic of research, due to the lack of an existing consensus on cyber-physical security metrics, and the need to establish standards in order to reduce risk to the point where buyers will begin to actually purchase such systems and commercial activity can accelerate further progress.

Among the many problems that require R&D focus, the following specific areas were noted for their difficulty:

1) Designing a high confidence system with humans "in the loop", i.e., detecting insider threats.
2) Achieving high confidence in dynamic, distributed systems subject to reconfiguration and interdependency.
3) Integration of ad hoc wireless systems into existing systems as an alternate line of defense (overlay network) against intruders.
4) Building security into our embedded system design processes, performance metrics, standards and certification procedures.
5) Recognizing software itself as vulnerable, and instilling "security self-awareness" at the lowest levels of embedded operation
6) Achieving on-line, real-time IT upgradeable embedded systems that evolve while retaining confidence, but without the need for downtime.

2.2.5 Research Strategy and Roadmap

Priority 1: Intrusion Detection and Self-Healing Systems

Roadmap:
0-5 years:
Intrusion detection for cyber-physical systems
Basic research in software self-healing methods and security-robustness concepts (science)
5-10 year:

Test beds for intrusion detection and self-healing
Component-level self-healing exemplars and focused demonstrations
Basic research on system-level security tolerance concepts, certification methods (science)
10-15 years:
System-level security-tolerant examples
Subsystem-level test bed demonstrations of component and system level self-healing systems.

Priority 2: Metrics
Roadmap:
0-5 years:

- Definition of Security process and performance metrics that complement existing throughput, reliability, and latency metrics for control and embedded systems and are consistent and compatible. Classification of areas of vulnerability.
- Characterization of the trade-offs between security and existing performance measures such as throughput, reliability and latency. Incorporation of physical security into traditional functional models of performance. (Science)

5-10 years:

- Models of embedded segments of process and utility industries that exhibit security tradeoffs with other performance metrics (e.g., in SCADA or DCS environments).
- Characterization of dynamically upgradeable metrics suitable for evaluation on operational systems (or perhaps shadow systems). (Science)
- Definition of practical devices and equipment for measuring security metrics.

10-15 years:

- Initiation of standards that incorporate new security metrics
- Initiation of certification methods for new security metrics that are compatible with existing standards for performance.
- Security demonstration test-beds and examples exhibiting improved security metrics relative to baseline (existing) systems.

Priority 3: Hardware/Software Co-Design for Security and Reliability
Roadmap:
0-5 years:
Identification of critical hardware and software security threats
Identification of specific, low-level "operate-through" and related device and software requirements based on most likely system level threats. (Science)

5-10 years:
Models of embedded sub-systems suitable for assessment of threat impact
Development of hardware-software co-design concepts suitable for enhanced security assurance (e.g., hardware-software combined encryption or authentication methods). (Science)

10-15 years:
Demonstration examples of secure modules or devices designed using hardware-software co-design concepts
Extension of modular- to system-level security.

Priority 4: Sensing for Situational Awareness
Roadmap:
0-5 years:
Identify sensing priorities (both hardware and software) for improved threat detection.
Explore geophysical situational awareness concepts (science)
5-10 years:
Demonstration of sensing systems that exhibit improved ability to detect (or accommodate) threats
10-15 years:
Combine sensing and geophysical systems to demonstrate use of mobile assets for early responders to isolate threat sources.

Priority 5: Formal Verification
Roadmap:
0-5 years:

- Develop mathematically precise definitions of security requirements (Science)

5-10 years:

- Extend existing formal reliability-availability basic research methods to apply to formal verification of security requirements. (Science)

10-15 years:

- Demonstrate prove-ably correct security-enabled hardware/software embedded components.

Lower Priorities
Roadmap suggestions:
Heterogeneous System Interconnections
5-10 years:
Develop system specification methods for heterogeneous components that allow the estimation of security, safety, reliability, and latency of systems built of such components. (Science)

Modeling large multi-scale phenomena
10-15 years:
Demonstrate use of multi-scale models to verify security performance of embedded systems such as utility distribution management or building system controls.

Security-capable Networks and Integration of Components
0-5 years:

Identify security and safety “forensics” requirements that allow for rapid threat assessment during or shortly after failures of systems (utility or process control systems) incorporating embedded components. (Science)
5-10 years:
Explore the use of distributed intelligence as a means of enhancing safety and security of control systems (decentralization of information) (Science)
Examine new software version control methods suitable for rapid system assessment and regeneration in response to failed components. (Science)
10-15 years:
Explore generative modeling and automatic model adaptation to accommodate security threats or system failures.

Training of researchers who are knowledgeable in both cyber-security and embedded systems is a high priority for being able to carry out this research agenda; one participant suggested that only about 100 persons worldwide are expert in this area. Better training of embedded systems developers in the software aspects of present IT processes, and secure IT processes (based on non-embedded systems) should be part of this training program. Notions of model-based design are not common outside of the controls field, but hold significant benefit for future secure applications. Finally, there is a need to train plant operations engineers and IT personnel in new methods and processes for secure IT management of embedded systems.

2.3 Design Methodologies and Tools

2.3.1 Introduction

The technology of networked systems is increasing in power, flexibility and is enabling new applications, however, the risks associated with product development that are measured by development time, cost and product robustness must be addressed through the creation of design flows and tool chains that mitigate the risk and enable widespread use of the technology. The tools must in particular be readily available and must have been developed with the goal of assisting engineering teams whose members have background in different disciplines (automatic control, communication networks, and hardware design, software programming). In addition, there must be a concerted effort to develop a workforce that is capable of designing networked embedded systems and an associated R&D effort to bring the different disciplines together. There is urgency to accomplish this R&D initiative from multiple directions including future market pressure as well as ensuring that the United States competitive position remains in the area of IT. There are significant risks to the deployment of both current and new products that utilize networks in both the commercial and military sectors, for example, the security of the critical infrastructure systems controlled through SCADA networks which specifically include particularly the nation's water and energy supply.

The roadmap elements identified for the development of design methodologies and tools fall into four categories. The first, design methodologies, addresses the need to overcome the complexity of networked embedded control systems (NECS) through the development of abstractions and design flows that are built around new models of computations and compositional rules. The intent of design methodologies is to guide the development of NECS in a structured manner. The second area on the technology roadmap is that of tools that support the design flow and here the need is very broad and urgent: the development of modeling tools for domain-specific applications; the development of multiscale modeling tools that can capture essential dynamics of both the physical and IT system; and the development of analysis methods such as graph-theoretic techniques that help identify the inherent structure of the problem and capture the probabilistic behavior of the overall system to assess performance in tractable ways. The next on the technology roadmap is that of developing foundational theory for NECS that bridges traditional and especially academic disciplines and addresses new elements that networked systems highlight. These elements include developing novel approaches for control architecture selection and control design that exploit the network features, how to represent and control numerous sources of uncertainty to enable robust product development and new approaches to compose varying models of computation. The last area on the technology roadmap concerns developing the talent base and also developing testbeds that can be used to focus multidisciplinary teams and also to mature key technologies by stressing novel architectures, algorithms and hardware implementations. There is an urgent need to structure R&D programs and the consequent influence on education specifically to build the technology roadmap for NECS.

2.3.2 State of the Technology

The current product development processes that exist for networked embedded control systems currently are carried out using a suite of design methodologies and tools. The existing infrastructure is enabling some nascent development and in this sense the infrastructure is providing value - albeit in an ad hoc and “borrowed” manner from existing design approaches. The methodologies and tools have been adapted from existing areas and while some design flows and tools support the design there are significant gaps.

The capabilities that currently exist that are important to highlight include the overall product development process, the use of modeling and the use of design automation tools.

- Existing capability in product development. It is important to realize that there are existing - and an increasingly large number - of networked embedded control systems. The range, as mentioned earlier, across market segments in automotive, building and industrial automation sectors speaks to the ability to field product. The existing capability includes product development processes for hardware and software development and robust product design - albeit often at productivity and risk levels that impede widespread deployment.
- Existing capability in behavioral modeling and simulation. The design of NECS currently uses various modeling approaches to capture requirements, to evaluate architectures and to quantify sensitivities. The tools support functional modeling of the physical system as well as some limited modeling of the IT infrastructure. Successful uses of models though are limited to “small” applications and the approaches currently do not scale to full system level descriptions or to “industrial scale” models that capture the entire range of interactions at the length and time scales that are necessary to reduce risk and have confidence in the robust operation of the deployed systems.
- Existing capability in design automation: code generation and testing. There are tools that are available to automate areas of the design flows especially in the area of automatic code generation from models of the system functionality and the testing of the resulting code for correctness. These methods are critical for productivity and to remove the “artisan” nature of embedded system deployment, however, the full range of automation tools for NECS are not available.

2.3.3 What we can’t do well

Networked embedded control systems (NECS) are increasingly being developed and deployed across a wide range of applications including the automotive sector, the oil, gas and industrial automation sectors and the commercial building sector involving automation and communications. The increased performance measured by new functionality that can come from the deployment of NECS range from increased monitoring and situational awareness - creating new functionality to make information available - to new control modes - using the information to alter the behavior of the networked system - that taken together offer compelling cases that can respond to both market pressures and to areas of national need. However, the technology and the development tools for NECS to enable widespread deployment are simply not mature.

There is a lack of design methodologies and tools to support the development process and as a consequence opportunities for deployment are not being seized except by the most adventurous industrial players, and in all application areas there is at least a lack of efficiency in the development process. As market and national needs continue to be manifested there is urgency to make R&D investments in the areas of design methodologies and tools so that products and services from NECS will be realized in the commercial and military sectors.

The nature of the challenge is to support product development through the introduction of design approaches and associated tool chains. The measures of success must be to decrease development time and cost, to drive down the risks of fielding novel products through robustness guarantees and to increase the overall ability of the industry to innovate through availability of a skilled workforce.

The need to develop methodologies and tools is urgent. A number of product failures have been associated with networked embedded control systems that illustrate the barriers to widespread development without systemic and sustained investment in this critical area. Examples include:

- Automotive recalls due to software
- Airplane crashes (e.g. Airbus 1994) due to pilot override of software
- Lack of competitive position due to inability to adapt to changes (manufacturing, food processing, semiconductor applications)
- Product development delays due to taking bad decisions and the inability to estimate project duration/complexity
- Lack of requirements management and consequent market misses and development turnbacks

2.3.4 R&D Challenges

Design methodologies and tools, while being deployed in some limited areas with success, are not widely used and do not cope with design issues in NECS. The purpose of this section is to highlight the limitations in current design practices to motivate the roadmap elements that are recommended for investment. A separate section highlights the science that is needed in the area of NECS for a foundational understanding of the field upon which engineering practice can be built.

Networked embedded control systems offer some new and significant challenges. The purpose of this section is to isolate some key areas where foundations are lacking and require significant attention from the R&D community at large - academia, government laboratories and industry - to address. The following are significant challenges in NECS.

- A challenge is to extract information from legacy software so that legacy systems can be used in new design flows and new features built from legacy systems considered as subsystems of new networked systems.
- A challenge is to develop design methodologies and tool chains to carry out joint designs of the physical and information technology subsystems where no “separation principle” of physical and network time scales exists. The lack of

such separation dramatically alters the approach to design and increases the complexity and scale far beyond the current tool infrastructure.

- A challenge is to address the time scale of design as well as the current deployment practices by making design tools part of the functioning infrastructure and dealing with continuous change. The intent is to have the design tools in place in the functioning system - counter to the current design paradigms - to enable maintenance and upgrades.
- A challenge is to address industrial-scale system-level models with the heterogeneity and scale that are found in application through development of layers of abstractions, tool chains that connect the layers and multiscale modeling approaches that capture the underlying dynamics at the appropriate scales to enable the stages of design (conceptual, detailed, ...) to seamlessly be connected.

- A challenge is the ability to deal with the complexity of designing large distributed concurrent systems made of tightly interacting components that operate concurrently. In particular, assumptions like the synchronous paradigm, which simplify the design of hardware systems and embedded software and enable the partitioning of design elements to subteams to structure the design process, are not necessarily matched by the distributed nature of cyber systems.

The R&D Areas: The Limitations to Current Practice

Design Flows

A number of issues exist in the approach to design that do not address the overall complexity of NECS and limit the development speed and cause undue risk and cost in the product development process.

- It is not currently possible to formally make the network part of the control design process - coupling dynamics and semantics of what are two currently separated design elements.
- It is not currently possible to explicitly deal with unreliable hardware and consequent uncertainty arising from the interconnection of unreliable subsystems.
- It is not currently possible to effectively deal with system specifications at different levels of abstraction to enable reuse of the development efforts and to enable design teams to collaborate and communicate across disciplines.
- It is not currently possible to effectively deal with asynchronicity as an underlying design element.

Tool Chains

The purpose of design methodologies are intended to form a structured process for development of NECS. It is critical that tool chains that can implement the steps in the process be developed that can both support the methodology in a structured fashion but also provide automation for increased productivity. In the area of NECS there are significant gaps in current practice.

- There do not exist advanced compilation technology (model building) that are robust and viable at industrial scale in terms of the heterogeneity and scale that are needed to be supported.
- There do not exist support tools for maintenance and modification after initial construction of NECS.
- There do not exist model checking technologies that are scalable for industrial-scale deployment.
- There does not exist the capability to compose semantics of subsystems and effectively deal with the heterogeneity of subsystems that are found in NECS and especially in the cyber-physical interfaces.
- There does not exist the ability to capture fundamental limits of performance for large networks and in particular there is a lack of abstraction levels that support this analysis.
- There does not exist a set of domain specific modeling languages that enable the capture of requirements and enable the subsequent automation of design flows.

Education

Education in the area of NECS must address the fundamentally multidisciplinary nature of the technology and applications.

- There do not exist approaches for R&D that address the multidisciplinary nature of NECS. The need to address all elements of a design methodology from specification development, domain expertise and physical modeling, architecture development and hardware and software implementation must all be addressed in R&D programs that address design methodology and tool issues.
- There do not exist testbeds that are specifically created within application domains to stress research environments that are developing NECS architectures, algorithms and research implementations.

- The testbeds must be designed for flexibility but also to focus research teams and to mature technology to lower barriers for industrial transition including the transition of people from R&D environments as well as tools to enable design teams to increase productivity and deploy products.

Measuring Success

The limitations of current practice impede the ease of developing and deploying NECS in various sectors. The purpose of this section is to outline high-level milestones that should serve to measure the progress and success of R&D efforts in NECS.

- Ability to transfer design specifications to actual interoperability of networked subsystems, components and modules.
- Ability to specify and to guarantee quality of service of the overall system - and require no testing - in the face of uncertainty of the environment, as well as physical and network elements.
- Ability to identify the critical parameters of the system directly from specifications utilizing models.

- Ability to co-design the physical and network elements of the overall control system.
- Ability to deal with lack of assumptions of synchronous behavior of the physical and cyber subsystems both at the conceptual design phase and implementation phases of design. The ability should enable the partitioning of design elements to subteams to structure the design process.
- Availability of domain specific languages to enable design teams to work at higher levels of abstractions.
- Graduate curriculums and students with skill sets (through significant public-private partnerships) that have developed and demonstrated on testbed R&D projects.

2.3.5 Research Strategy and Roadmap

A roadmap for R&D in the area of networked embedded control systems can be organized around the elements of (a) design methodologies, (b) tool chains, (c) theoretical foundations and (d) educational elements including testbed development and technology transition.

Design Methodology Development

- Requirements capture and tracking for networked embedded control systems.
- Development of abstraction and refinement processes between various levels of abstraction layers for networked embedded control systems and specific use cases for key application areas.
- Development of methodologies to address asynchronous/synchronous networked subsystems.

Tool Chain Development

- Stochastic modeling and analysis tools.
- Multiscale and dynamic modeling and design tools.
- Domain-specific metalanguages to enable capture of required behavior and subsequent design automation.

Development of Theoretical Foundations

- Theories for the composition of subsystems across multiple models of computation.
- Theories for NECS control architecture, design and implementation that include novel capabilities and enable fundamental limits of performance analysis.
- Theories for uncertainty representation and control in NECS.
- Theories for security: effective dealing with attackers.

Educational Development

- Testbed definitions, creation and demonstration.
- R&D program organization stressing teams and technology maturation.
- Industry-university partnerships.
- Redesign of education curriculum in signals & systems, control and embedded systems to focus on modern system science.

2.4 Enabling Technologies

2.4.1 Introduction

This section identifies the fundamental advances in technology that are required to realize future networked embedded control systems, and presents a preliminary research roadmap for realizing these key enabling technologies. Success in fielding future systems requires advances across multiple domains, spanning devices and hardware, sensors and actuators, communications, advanced software, security, system integration tools, control theory, man-machine interface, system operation principles. It is important to realize that ultimately we are dealing with a systems engineering problem. While individual enabling technologies will be identified, the most difficult problems are those that concern the total impact of different technologies as a whole on system performance. Thus advance in any one particular enabling technology must be viewed in the context of how it will affect the tradeoff decisions and engineering compromises inherent in the design of complex systems.

2.4.2 State of the Technology

Modern society cannot function properly without an adequate level of service that is provided by today's networked embedded control systems. To the extent that today's technology is sufficient to deliver essential services such as electric power, transportation, water supply, our technology base is adequate to sustain the day-to-day working of a modern society. However, what we can apparently do well now is predicated on the assumption that the demand on our infrastructure will not be stretched to its limits. It is more accurate to say that our technology base for embedded control systems may be adequate for today's needs only if we operate well within the system capacity envelope and that we discount scaling-up issues that take into account factors such as terrorist attacks and natural disasters.

Hurricane Katrina gave us a vivid demonstration that we must do a lot better in strengthening our technology base to cope with large-scale dislocations. In the area of R&D, we are quite good at demonstrating the potential of new technologies only at the level of small-scale demos**.** *We are quite good at the modeling, analysis and evaluation of small-scale systems, but we have no reason to believe that our technology base can be scaled up to meet the demands of the networked embedded control systems of the future, especially when operating under duress.* For example, as our population ages, we shall be losing some of the critical expertise for managing complex process control plants, thus raising safety issues should emergencies arise. Tele-presence technology can in principle be used to allow experts to interact with workers at the site of a process control plant to defuse a dangerous situation. However, our current technology base for virtual immersion is not at a level that can be scaled up to allow an expert to perform as effectively from a remote location as does on site.

What we can do well with centralized systems where short physical distances incur short communication latency may not work at all in a distributed environment where long physical distances incur long communication latency and the control system also has to deal with resource contention issues. The result can be system instability.

2.4.3 What we can't do well

In general, we have no reason to believe our current technology base can be scaled up for cyber-physical systems of the future. We can identify a number of specific things that we cannot do today because of unfilled needs in enabling technology. We give some examples of the specific needs below:

1. **We need a much better capability in collecting geographically distributed data that can be aggregated to produce a global picture of the system state in real time**. For example, in the power industry, it is well known that power generation and routing can be performed much more effectively if we have accurate phase data from the geographically dispersed power generators. This will only be possible if the deployment and integration of PMU (phase measurement units) can be scaled up on a nation-wide scale and phase data can be made available in real time to yield a global picture of the national power grid.
2. **We need to improve our technology base to enable anticipatory control.** An example of anticipatory control is the proactive routing of traffic in the power industry. Substantial savings can be realized by routing power around the trouble spots in advance of an expected disruption such as a major storm. The current practice is reactive; remedial action is taken after trouble has already developed and economic loss becomes inevitable. Alternatively, the routing decision could be made in anticipation of severe weather conditions that might bring down power lines. To make anticipatory control happen, technical advance is required beyond the deployment of current information technology. Ubiquitous access to information, supported by cross-platform interface for both wired and wireless devices is a prerequisite to the automation of control decisions. A coordination network that employs both push and push technologies and can reason about and identify the adverse impact of over-the-horizon events is required to fully realize the economic gains from anticipatory control.
3. **We need to demonstrate the ability to scale up the deployment of embedded sensors to enable large-scale real-time control.** While we are making steady progress in sensor fabrication technology, there is a wide gap between small-scale experimentation of sensors and wide-area deployment of sensor networks, and between the deployment of sensor networks and sensor-actuator chains to enable real-time control. Current experimentation of sensor networks by researchers is often small-scale and severely constrained by power limitations. One recent attempt to instrument the Golden Gate Bridge with structural sensors was severely handicapped by the limitations of current battery technology. The fact is that we cannot extrapolate the results from small-scale experiments to nationwide-scale applications. Before we can instrument, for example, the major transmission lines nationwide with embedded sensors, we need a reliable way to evaluate the scalability of the deployment strategy for millions of devices. We need the capability to manage millions of devices in the field, including possibly the need to perform software upgrade and system reconfiguration. An even more serious problem is the management of millions of actuators whose actions, if uncoordinated, can cause loss of lives and property. We do not know how to scale up to control systems that involve millions of actuators and

must deal with long latencies and imperfect information that are likely to be commonplace in large-scale sensor/actuator networks.

4. **We need to ensure the trustworthiness of data capture, transmission and retention.** Modern communication theory has enabled the reliable transmission of data by sophisticated application of coding theory. However, loss of trustworthiness of data can occur in sensor/actuator networks because of inadequate handling of resource constraints such as worst-case processing speed, network overload, memory overflow. In the protection of our critical infrastructures such as the power grid, we need to ensure that the data captured from sensors and delivered to actuators cannot be corrupted by a malicious attacker or by the mismanagement of computing, storage and communication resources. For wide-area active control systems, failure to deliver sensor/actuator data in a timely manner compromises system integrity. This can happen because of malicious actions (e.g., DDoS attacks), software/hardware failures or improper scheduling of resources.
5. **We need modeling, requirements capture and system engineering tools that span both cyber and physical domains.** We need a tool chain whose components have explicit and well defined semantics to enable the modeling, design, implementation and evaluation of systems that are comprised of components with:
    - **a.** Continuous and discrete-time dynamics
    - **b.** Mixed synchronous and asynchronous coordination schemes
    - **c.** Multiple temporal and spatial scales
    - **d.** Hierarchical structures

   The tool chain must be integrated so that the effect of a change at any stage can be and must be automatically reflected at other stages. We also need tools to resolve conflicting design objectives.
6. **We need to vastly improve our current software infrastructure for supporting large-scale and distributed real-time active control systems.**
    - a. Current RTOS (Real-Time Operating Systems) provide only limited support (e.g., RMA scheduling discipline) for timeliness requirements. Better support is needed for end-to-end timeliness requirements, jitter constraints, I/O scheduling, real-time garbage collection, etc.
    - b. Networking and data transmission support for embedded and wireless applications need to be more robust and secure, preferably with built-in security and robustness guarantees.
    - c. Current middleware is too heavy for embedded real-time applications. Light-weight middleware is needed.
    - d. Better isolation mechanisms are needed to facilitate reconfiguration and self-healing, e.g., light-weight and hardware-supported resource virtualization technology.
7. **We need to better understand the properties of large-scale distributed coordination schemes**. Centralized processing is not possible for large-scale wide-area active control systems inasmuch as there is too much data to process and physical distances incur excessive latency for maintaining system stability. It will be unavoidable to distribute "intelligence" into the network and hence a need exists for efficient and robust distributed coordination schemes. Coordination schemes that are fine-tuned may be too fragile to withstand changes in the environment.

8. **We need to solve problems at the enterprise level by tying in process-level control strategies to enterprise level concerns.** For example, layer 3 of the OSI model does not adequately address cross-enterprise needs for performance and security. On the other hand, layer 1 or 2 of OSI model (e.g., as provided in NGI IPv6) is not used adequately in practice. The problem is not purely technical. Successful adoption of technology innovation requires better understanding of the psychology of technology adoption. A possible solution to ease technology transition is virtual reality-based training. However, our technology base is still inadequate for high-fidelity virtual reality-based training in real time for most active control systems.

2.4.4 R&D Challenges

In the previous section, we list a number of needs that are currently unfilled. In identifying these needs, we also state the specific challenges that must be met by a well considered R&D program so that the results from R&D can be used to field the beyond SCADA networked embedded control systems of the future, as listed below:

1. Build a better capability for collecting geographically distributed data that can be aggregated to produce a global picture of the system state in real time.
2. Improve our technology base to enable anticipatory control.
3. Demonstrate the ability to scale up the deployment of embedded sensors to enable large-scale real-time control.
4. Ensure the trustworthiness of data capture, transmission and retention.
5. Invent modeling, requirements capture and system engineering tools that span both cyber and physical domains.
6. Improve our current software infrastructure for supporting large-scale and distributed real-time active control systems.
7. Understand the properties of large-scale distributed coordination schemes.
8. Solve problems at the enterprise level by tying in process-level control strategies to enterprise level concerns.

Each of the above challenges invites multiple solution approaches and substantial experimentation work. A successful response to any of these challenges constitutes an enabling technology for the beyond SCADA systems of the future. It is impossible to cover all the solution approaches to these hard problems. In the following, we mention a number of technologies that are deemed to be especially promising in their potential payoffs and can thus be regarded as examples of disruptive technologies.

- Agent technology.
  Agent technology lends itself to the seamless distribution of "intelligence" into a distributed control system. Agents are software entities that, like Web crawlers, can collect information from the network connecting the distributed control system. Agents can form teams to continually monitor the state of the system under control and perform goal-directed tradeoffs between conflicting goals such as profit versus delivery constraints in a power grid. Agents can also be used to implement

application-specific query engines that allow the human operator to ask declarative (i.e., non-prescriptive) questions about the state of the system.

- "Eternal" O.S. and run-time system.
  The motivation of an "eternal operating system" (EOS) is that we do not and may never have provably reliable system software (e.g., see Windows and Linux) for high confidence systems and applications. The goal of an EOS is to make operating systems be able to cope with failures by:
  1. Never going down (self-regenerative, self-recovery, self-adaptive, self-updating, self-synchronizing, etc)
  2. Adapting to new threats and update itself
  3. Adapting to new operational environments, recover from loss of components, and grow by incorporating new hardware/software components
  4. Supporting end-to-end quality of service

- Trustworthy sensors.
  The motivation of trustworthy sensors is that we have concrete denial of information problems that indicate the inevitability of the trustworthiness problem: email spam (including phishing), web spam, blog spam, etc. The assumption that sensor output will not be maliciously manipulated will not hold once someone figures out how to profit from the manipulations. Trustworthy sensors must:
  1. Bypass the limitations of known methods. Current reliability and fault tolerance methods typically cannot handle adversarial (malicious) input manipulations that adapt to defense mechanisms. See the evolutionary history of spam content as concrete example.
  2. Incorporate new research in information filtering and integration to distinguish good sensor data from fabricated data and be able to adapt to changes in the environment. Automatic methods to maintain trustworthiness of data may include self-calibration from system-level analysis and contextual analysis.

- Tool chain that can close the process loop from specification to deployment.
  While current model-based design tools have helped ease the engineer's job in transforming controller designs into code, an essential problem remains which is the interoperability of the tools in the tool chain. One way to ensure interoperability is to force all the tools to share a common semantic base that is used to define the semantics of the tool's operations. This implies that the input and output for each tool are objects that must lend themselves to semantic analysis vis-à-vis the common semantic base. However, this approach might be too heavy-weight in that the tools in a tool chain often deal with different types of properties of the design artififact and it is unnecessary for the tools to share a common semantic base as long as they are in some sense compatible, e.g., a mapping exists that can propagate the effective of a design decision by one tool to another. Tool chains that can close the process loop in this way will help lay a formal foundation for a science of design for the beyond SCADA systems of the future.

- New human-machine interface technology.
  We believe that any technology that can significantly improve the effectiveness of the human operator in the control loop is a potential disruptive technology. An example is the WICAB TDU (Tongue Display Unit). The TDU allows the human operator to use his tongue to "see" an image faster than using his eyes because the biological sensors on the human tongue can respond faster than the human eye. Another example is the haptic glove that allows a human operator to manipulate physical objects from afar with the sense of touch. A high-fidelity haptic glove (which does not currently exist) will enable a human expert to manipulate complex machineries remotely, e.g., performing repairs from a safe distance in an emergency.

- Advanced hardware technologies.
  1. Interface to and control of nano, organic and quantum sensors
  2. Real-time holographic virtual interface for facilitating tele-presence
  3. Power management, storage device with ultra small form factor, ultra high energy density
  4. Quantum clock synchronization devices
  5. Multi-core processor + cell phone (multiple radios) capability

2.4.5 Research Strategy and Roadmap

Many of the issues that must be resolved in order to field the networked embedded control systems of the future are system integration problems that cross current academic disciplines. For control systems, engineering modeling and design must take into account implementation artifacts (e.g., computer networks that incur significant and sometimes unknown delays and jitter) that cannot be easily abstracted away. For computer science, the physical aspects of the control system and the plant impose an execution environment where there is at least one process (the environment) that is not directly under program control. We believe that in order to make progress, it is important to encourage cross-domain research teams that engage both application and control engineers and computer scientists. Apart from cross-fertilization, we hope that this research will result in some kind of "separation principle" that will enable a division of labor so that the control engineers can focus on the control problem and the computer scientist can focus on software/hardware architectural issues. For example, this might take the form of an abstract model that can be analyzed by control engineers for control properties and by computer scientists for implementation requirements in a distributed and networked environment. Such a model might even be embodied by a programming language that can be compiled into code. Of course, the meaning of compilation might have to be extended to include engineering analysis. With the complexity of the networked embedded systems, the bottleneck to progress may ultimately be the dearth of designers who have sufficient expertise in both control systems and software/hardware engineering. It is difficult enough to be expert in one of the two domains, so it is crucial for research to codify the knowledge required for control system design as well as hardware/software/networking implementation in as simple a language as possible, such that the required training for producing competent design is minimized.

In the following, we attempt to set up milestones for a research program in beyond SCADA networked embedded systems. Rather than giving quantitative measures that at this point are probably meaningless, we try to characterize progress by what a novice designer (one with a university education), expert can be expected to do in the future.

	3-5 years	6-10 years	over 10 years
Scalability	difficult for expert	routine for expert	routine for novice
Survivability	difficult for expert	difficult for expert	routine for expert
Embedded intelligence (agent technology)	routine for expert	routine for novice	
System infrastructure (closed loop tool chain)	difficult for expert	difficult for expert	routine for novice
Security	difficult for expert	difficult for expert	difficult for expert

Appendix 1. Workshop Agenda

National Workshop on
Beyond SCADA: Networked Embedded Control
for Cyber Physical Systems

Agenda
8-9 November, 2006

NOTE: Meeting rooms are listed at the end of the agenda.

Wednesday, November 8

7:30	Registration and Breakfast
8:15	Introductory Session (Session Chair: Shankar Sastry) NITRD (Simon Szykman), NSF (Helen Gill), NIST (Albert Wavering), NSA (Brad Martin)
9:00	Keynote Speaker (Session Chair: Shankar Sastry) Jeff Potter Security and IT Integration Manager Emerson Process Management - Rosemount Division
9:30	Questions and discussion
9:45	Keynote Speaker (Session Chair: Shankar Sastry) Anoop Mathur Technology Manager, Wireless and Embedded Controls Honeywell Labs
10:15	Questions and discussion
10:30	Break
10:45	Position Paper Briefs & Discussion I (Session Chair: Bruce Krogh)
12:15	Lunch
1:15	Charge to Working Groups (Sally Howe)
1:30	Working Group Sessions I
2:45	Break
3:00	Position Paper Briefs & Discussion II (Session Chair: Bruce Krogh)
5:30	Adjourn
6:00-7:00	**Reception**

Thursday, November 9

7:30	Registration and Breakfast
8:15	Keynote Speaker (Session Chair: Marija Ilic) Ronald Ambrosio Sr. Technical Staff Member, Event-Driven System & Relationship Manager, Energy & Utilities IBM TJ Watson Research Center
8:45	Questions and discussion
9:00	Keynote Speaker (Session Chair: Marija Ilic) Joe Weiss KEMA, Inc.
9:30	Questions and discussion
9:45	Break
10:00	Position Paper Briefs & Discussion III (Session Chair: Bruce Krogh)
12:15	Lunch
1:15	Working Group Sessions II
3:00	Break
3:15	Working Group Reports
4:00	Adjourn

Meeting Rooms

Plenary Sessions: Grand Station Ballroom III-IV (first floor)

Working Groups (all on second floor):
WG 1: Haselton I
WG 2: Haselton II
WG 3: Stoops Ferry
WG 4: Edenburg

Lunches: Reflections (first floor)

Reception (Wed., 6:00-7:00 PM): Waterfront (first floor)

Working Group Chairs Meeting (Friday, 8:00AM): Edenburg

Appendix 2: Workshop Participants and Working Groups

Working Group 1: Emerging Capabilities

R. Ambrosio (IBM), S. Amin (UC Berkeley, scribe), M. Crow (Univ. of Missouri, Rolla), M. Hartman (GE), M. Ilic (Carnegie Mellon Univ.), S. Isovitsch (Carnegie Mellon Univ.), T. Kaga (Toyota), I. Krüger (UC San Diego), B. McMillin (Univ. of Missouri, Rolla), G. Manimaran (Iowa State Univ.), A. Mathur (Honeywell), W. Milam (Ford), J. Moyne (Univ. of Michigan, Ann Arbor), J. Nash (As One Technologies), A. Ravindran (Univ. of North Carolina, Charlotte), T. Samad (Honeywell, lead), P. Sauer (Univ. of Illinois, Urbana-Champaign), Al Wavering (NIST)

Working Group 2: Security, Safety, and Certification

Rajeev Alur, Madhukar Anand, Terry Benzel, Brian Isle, Anthony Joseph, Clifford Neuman, Ernie Rakaczky Kurt Rohloff, William Sanders, Peter Sholander, Joe Weiss, Yuan Xie, Hakan Yazarel,

Working Group 3: Design Methodologies and Tools

Alessandro Abate, Panos Antsaklis, Andrzej Banaszuk, Ken Butts, Luca Carloni, Michael Feldman, Alberto Ferrari, Helen Gill, John Koo, Jerrold Marsden, Igor Mezic, Pieter Mosterman, Alberto Sangiovanni-Vincentelli, Shankar Sastry, Dawn Tilbury, Hakan Yazarel, and Yuan Xie

Working Group 4: Enabling Technologies

Coordinator: Aloysius Mok, John Koo, Alberto Cerpa, Bharat Joshi, Wayne Manges, Bruce McMillin, William Milam, Calton Pu, Wei Ye