ADCHEM 2006

International Symposium on Advanced Control of Chemical Processes
Gramado, Brazil – April 2-5, 2006

SYSTEM ARCHITECTURE FOR PROCESS AUTOMATION: REVIEW AND TRENDS

Tariq Samad*, Paul McLaughlin, and Joseph Lu*****

**Honeywell Labs, 3660 Technology Drive, Minneapolis, MN 55418, U.S.A.,*
tariq.samad@honeywell.com
***Honeywell Process Solutions, 1100 Virginia Drive, Fort Washington, PA 19034, U.S.A.,*
paul.f.mclaughlin@honeywell.com
****Honeywell Process Solutions, 2500 W. Union Hills Drive, Phoenix, AZ 85027, U.S.A.,*
joseph.lu@honeywell.com

Abstract: New developments in information technologies are radically transforming process automation. Their impact and benefit derive both from these technologies individually and from their convergence in new system architecture concepts. This paper reviews how process automation system architectures have evolved and discusses future trends. We draw an analogy between the synergistic new technologies being developed today and the technology landscape of the early 1970s—characterized by the near-simultaneous appearance of microprocessors, communication networks, CRT displays—that resulted in the first DCS system (the Honeywell TDC2000). Emerging technologies highlighted include wireless, embedded devices, service-oriented architecture, and application infrastructures. *Copyright © 2006 IFAC*

Keywords: Architectures, process control, process automation, systems engineering, distributed control systems (DCSs), communication networks, information technology

1. INTRODUCTION

A process plant is a complex, multifaceted entity, a structured organization of physical elements, operated for economic and other criteria that are often industry-specific, and with a number of different stakeholders who can affect and/or are affected by its operation. Critical to the operation of the vast majority of plants today is an automation system that performs control and other advanced functions including, but not limited to, optimization, scheduling, and planning. The automation system ensures that appropriate parameters are measured, operational situations analyzed, more profitable opportunities explored and control actions calculated and taken, plant personnel kept informed and their knowledge and capabilities exploited, abnormal situations identified and addressed, and business processes integrated. The components and devices of the automation system perform functions that are essential for safe and efficient process operation, but it is the system architecture—the logical organization of the components and associated infrastructure—that often dictates choices of components and determines key system performance features such as reliability, capability, throughput, scalability, and cost. The system architecture also dictates how well applications are integrated, deployed and supported throughout their life-cycles, and in what manner the application functionality is delivered. A major theme for a system architecture is thus (1) to devise infrastructures, services, components, and their organizational schemes for best delivering the automation functions including advanced application capabilities; and (2) to integrate—to cohesively combine seemingly disparate components into an effective and consistent whole. It's the architecture that makes a system more than the sum of its parts.

System architecture is a hard thing to define crisply, let alone discuss in any depth. It is not a component, even an abstract one. It has enormous impact on how, and how well, we operate our plants, but its "emergent" nature is somewhat at odds with the research community's typical focus on specific technologies and their applications. Individual technology developments relevant to process automation are often discussed in depth, but we seldom examine how multifarious developments can result in the synthesis that is architecture.

This paper focuses on architecture for process automation systems. We first discuss the key issues related to process plant operations that are affected by automation system architecture. Next, we briefly review the history of process automation with specific reference to the development of distributed control systems (DCSs). Several technology developments are likely to dramatically transform process automation architectures in the near future; we highlight some of these developments. We conclude with a comment on the connection between system architecture and research topics in the controls community. Readers seeking a broader architectural perspective on the process industries as enterprises may find the Purdue Enterprise Reference Architecture of interest (www.pera.net).

2. THE IMPACT OF ARCHITECTURE

Architectural choices can profoundly impact how well we manage and control industrial processes—indeed the scale and complexity of the typical plant elevates the importance of architecture. Here we briefly discuss the connection between system architecture and each of several "critical to quality" (CTQ) criteria.

2.1 Applications Capability

The number, the variety, and the sophistication of advanced software applications for process automation continue to grow, and the automation system architecture determines how rapidly and cost-effectively they can be developed, implemented, and maintained. Architecture impacts applications through functions and features such as support for data types, interprocess communication mechanisms, on-process migration, real-time scheduling policies, and componentization and interoperability of modular blocks.

Four classes of avanced applications can be differentiated:

- *Process effectiveness* applications provide better control/optimization, increase throughput, reduce operating cost and waste, improve product quality, and ensure regulatory compliance.
- *Asset effectiveness* applications predict and pre-empt asset malfunctions, reduce maintenance costs, prevent asset decay (e.g., corrosion), and enhance asset security
- *Business effectiveness* applications respond to seasonal change or volatility in markets. They optimize what to produce, when, and in what quantity.
- *People effectiveness* applications improve operator proficiency, reduce/avoid unplanned capacity loss, implement/audit best work practices, and turn data into actionable information or knowledge for plant staff.

2.2 Reliability

It is inevitable that automation components—sensors, transmitters, actuators, displays, panels, wires, routers, etc.—will fail or require offline maintenance. Given the quantity of automation equipment in a plant, in fact, it is a virtual impossibility that every piece of equipment is functioning correctly at any instant. Yet we expect—and generally realize—high levels of process uptime. In large part, this is because of reliability features that have been designed into the automation architecture. A number of architectural approaches that can help improve reliability have been adopted. We note four here:

- Reliability can be achieved via redundancy—e.g., parallel, dual communication networks.
- Fundamental to system reliability is the ability to diagnose for faults and to annunciate these faults to both plant personnel and operational logs. An undiagnosed fault in a redundant element means that the availability of the solution is no better than having a nonredundant element.
- A distributed system (if appropriately designed) can improve reliability over a centralized system by collocating (or more closely locating) critical components. Distributed systems are not universally more reliable than centralized ones, however; the former can be easier to maintain and upgrade and synchronization and consistency issues are of less concern.
- Aspects of system architecture beyond the physical also influence reliability. Thus communication semantics in process automation include "failure" signals that are separate from the "values" being communicated. A failure signal can trigger automatic reconfiguration or promptly raise an alarm for an operator.

2.3 Lowest Total Installed Cost (LTIC)

LTIC is an important decision criterion for new and upgrade installation of automation systems. With proprietary automation systems largely giving way to open ones, plant owners and managers have many more supplier options. The cost and ease of integration can vary substantially among alternatives. LTIC includes the product cost itself, as delivered, plus the cost of installing it in the plant and configuring it so that it can be brought online and integrated with the process automation system. The automation system architecture affects both installation and configuration. Wireless is now widely seen as a game changer in industrial automation, principally because it removes the need to run wire to every new device—especially valuable for upgrades to existing plants where the cost to add wiring is prohibitive. For a typical sensor installation today, the wiring cost handily exceeds the cost of the component itself, so wireless-enabled devices can command a premium if the process automation system architecture is wireless-capable. Often power and communication wiring are separately installed or power may be available at the point of installation anyway, so substantial savings are to be gained even if the wireless capability on a device is only for communication.

In addition to the physical installation and software installation and administration, configuration is also required. Whether hardware or software or a combination of both, a device will have parameters, methods, and other settings whose appropriate values must be specified. In many cases this configuration will require the use of purpose-built tools. The software architecture of the automation system in particular can affect ease of configuration. Autodiscovery features, Web servers and Web services, and shared semantic models are some features that can reduce configuration cost—by enabling, for example, a high-school-graduate operator to do the job instead of a trained engineer, by enabling the configuration to be performed remotely, and by reducing the time required for it.

We're not quite there yet, but we can envision a not-too-distant future in which a new device can be physically plugged into a network and be automatically discovered by the system and auto-configured, with only minimal human supervision.

2.4 Maintenance and Migration

A large continuous process will remain operational for decades, and any maintenance to or migration of the computer and control system components must be done in an online operational manner. This necessity presents a significant technical challenge for the automation system design team and for the end-user.

Online modification of the configuration of the system is a key requirement. Examples of online modifications include adding or removing new sensors, actuators and their interconnection to the control and monitoring system; adding new basic, supervisory or optimization controls; adding or upgrading human-machine interaction (HMI) consoles; loop tuning; and modifying the alarm and event reporting schema.

Online upgrade of all or part of the core system or application software is also a fundamental requirement; software releases can occur much more frequently (.5 to 2 years) than process shutdown cycles (3 to 8 years). View or control of any loops cannot be lost during software and system migration. Online upgrade typically depends on redundant computer and control components. In general, a secondary node is loaded with new software and is manually commanded to become the primary, while the primary remains in a passive backup state, with the original system software but able to resume its prior role. This capability is referred to as "load and go back."

Another significant migration and maintenance challenge for current DCS providers is the increased use of commercial-off-the-shelf (COTS) components, such as personal computers, servers, and network switches and routers. It is assumed by DCS customers that the supplier will ensure that the initial set of components will operate together correctly; it is also understood that the vendor shall provide methods for the customer to upgrade and replace these components over time while maintaining consistent online operations. The pace of change and obsolescence in PC and network technology far exceeds that of traditional DCS "proprietary" hardware.

2.5 Real-time Properties

Ultimately, what distinguishes a process automation system from an office automation system is the former's connection with a complex, dynamic physical system, an industrial process or plant. The process automation system is required for accurately and conveniently monitoring, controlling, and supervising the operation of the process. Continuous processes pose particular challenges in this regard, in particular since the timing of measurements and actions will influence not just the timing of the process's evolution but its very nature. The right action but with timing off by a few seconds (or less) can be ineffective or potentially even disastrous.

Two timing properties that are especially crucial for feedback control are latency and jitter. Latency refers to end-to-end delays associated with communication, computation, and actuation. In general, the greater the latency the poorer the quality of control that can be achieved—actions can only respond to delayed measurements, not current ones. For disturbance rejection, excessive latency can result in larger disturbance perturbation and performance degeneration. Worse yet, there is no control design or tuning that can improve the rejection performance within the latency band. For discrete logic and/or safety control, long latency can result in a disqualification of the control system.

Jitter refers to the variability of latency measurement. For feedback control (or any discrete-time application) it is the end-to-end jitter that is important. Even if a required sampling rate is maintained on average, for example, jitter is undesirable. During design and simulation of feedback controllers, jitter is difficult to consider because almost all the formulas and theorems of discrete-time mathematics used in control design assume jitter-less sampling. If encountered in the online system, control quality can be significantly poorer than suggested by the simulation results.

2.6 Scalability

In that a distributed system has a large degree of variability with respect to how it is assembled and organized, it is imperative that the system be designed to ensure that overall performance targets, capacity limits, and topology deployments be considered in the up-front design. It is very difficult to scale a "small" system into a large system, and likewise, very difficult to scale a system designed to be “large” into a cost-efficient and feature-bundled “small” system. An example of a small system would be one intended to control one piece of process equipment, and be constituted with a single HMI station, a single controller, and a small quantity of I/O (conventional or fieldbus-based). An example of a large system would be one designed to cover a large refinery, including all process elements (both continuous and discrete), all human-machine operations, and all business management functions.

2.7 Security

Security would not have been considered a “CTQ” a decade or two ago, beyond simple user classification and authentication (e.g. keys), but times and technologies have changed since. Both cyber and physical security are now top-of-mind considerations for automation systems. Computers in plants are now connected to the Internet. In some cases “air gaps” may be designed between the process-connected and Internet-connected parts of the automation system; in other cases the protection is a firewall. Wireless is already being adopted for some applications. Fortunately we do not know of any major accidents caused by cyberattacks, but “successful” pranks and inadvertent unauthorized accesses have been reported.

Unauthorized physical accesses are also a topic of concern, and there is increasing interest in access security, biometrics, and video surveillance. Integration within one automation infrastructure is the desired goal, not only for reasons of complexity and cost but also because of the synergy possibilities. For example, in addition to a password, a biometric recognition device could provide a second, automatic check for automation system access.

3. THE EVOLUTION OF SYSTEM ARCHITECTURE FOR PROCESS AUTOMATION

Process system architecture has gone through dramatic transitions since automation began to be applied on a broad scale to process plants, and we can already anticipate future revolutions. In this section we provide a brief and selective review of significant developments in process system architecture. Much of this section is derived from Przybylski (1989), James and Weir (1989), and Dallimonti (1985). The specific examples we discuss are largely Honeywell products and solutions since those are the ones we are most familiar with.

Not only were the first process automation systems not computerized, they did not even rely on electricity. All sensing and control was done pneumatically, with small-bore metal tubing used to convey pressurized air through the plant. Sensors would transform a measurement to an air pressure, actuators typically employed metal bellows to transform pressure to mechanical movement, and control was implemented via pneumatic devices.

The pneumatic system continued to serve as the model even after electrification. Instead of air pressure, a 4-20 mA current became the signal representation, but connections (now with wires) were point-to-point and controllers were assembled from discrete electronic components such as operational amplifiers, resistors, and capacitors.

Digital computers started to be used in industrial control in the 1950s, when the Ramo Woolridge Company (the precursor of TRW) won a contract to develop a computer control system for a catalytic polymerization unit at the Texaco Port Arthur, Texas refinery (Moore, 2003). At first, the computer was used only for data-logging, alarming, offline efficiency calculations and operator guidance. Process control was still done the old-fashioned way, with analog equipment.

The first completely digital computer-controlled systems relied on a centralized CPU, although analog controllers continued to be used for backup for reliability. With the advent of DCSs digital control began to be widely adopted.

3.1 The First DCS

Honeywell’s TDC 2000 (see Fig. 1) is generally recognized as the first digital distributed control system. Its development started with an internal proposal process in 1969 and culminated with the announcement of the system on November 11, 1975.

The TDC 2000 was revolutionary in its adoption and extension of new technologies:

- Board-based small programmable digital computers were developed that could serve as multiloop process controllers. The TDC 2000 featured the first 16-bit microprocessor in a commercial product. Each controller had 8 outputs, 8 control slots, and 16 inputs. It contained 16K × 10 bits ROM for the firmware and ran at a frequency of 3 Hz—thus 24 loops/sec. Control strategies that previously required a central minicomputer, with attendant reliability issues, could now be implemented on microprocessors. Timing, communication, and scheduling problems were greatly ameliorated.
- A serial digital communication network called the “Data Hiway” was used to link the controllers, the operator interfaces, and computers. This network was a primitive but pioneering local area network (although the term had not been coined and no such commercial technology existed then) with redundant media. (At the release event for the TDC 2000, the reliability of the Data Hiway was demonstrated

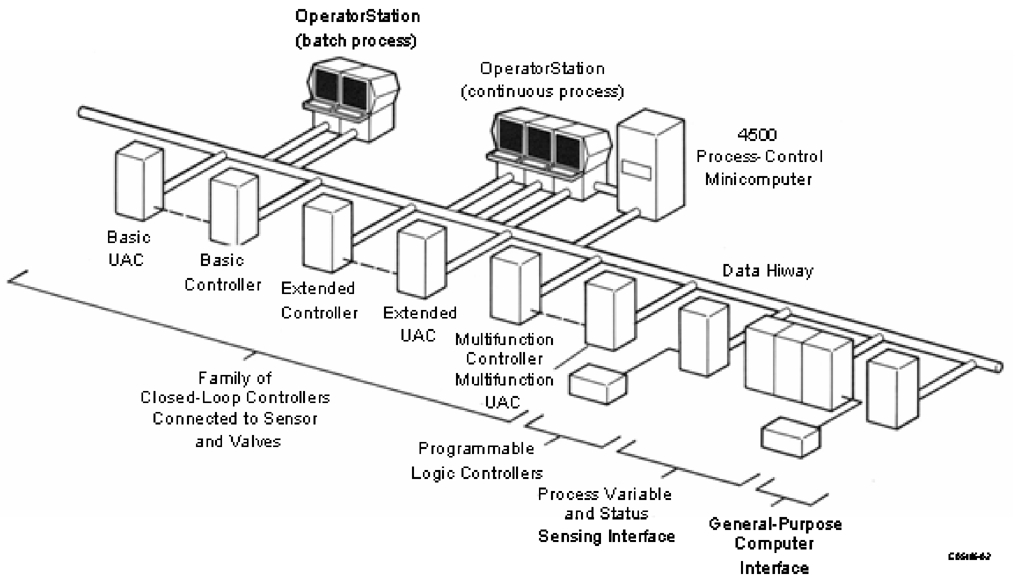
OperatorStation (batch process)
OperatorStation (continuous process)
4500 Process-Control Minicomputer
Basic UAC
Basic Controller
Extended Controller
Extended UAC
Multifunction Controller
Multifunction UAC
Data Hiway
Family of Closed-Loop Controllers Connected to Sensor and Valves
Programmable Logic Controllers
Process Variable and Status Sensing Interface
General-Purpose Computer Interface

Fig. 1. The system architecture for the Honeywell TDC 2000 distributed control system. From (Dallimonti, 1985)

by severing a cable with an axe. Because of the redundant design, the system continued to function normally.) Communication was at 250 kb/sec and controlled by a central scheduler. The replacement of point-to-point wiring with one digital network resulted in huge savings in installation costs—up to a million dollars for large jobs.

- Instead of large instrument panels, the TDC 2000 featured desk-size consoles with several CRT displays in the control room (Fig. 2). The CRT-based operator console allowed easy configuration of displays without any programming by end users and for the first time enabled the combination of process operations, alarms, and configuration into a console. The operator station was a precursor to the graphical user interface (GUI), which would later appear in the Apple Lisa and Microsoft Windows. The majority of customers were skeptical of the CRT console innovation when it was first released, and the operator stations could be ordered with analog displays for a more familiar look.

The initial TDC 2000 release included the basic controller, the Data Hiway redundant network, the basic operator station, and the supporting systems infrastructure—cabinets, power systems, battery backup, and a number of options called “analog modules.” The analog modules made the digital controller look like a traditional panel board and provided a level of backup capability. Later releases provided several enhancements, including the Data Hiway Port (DHP) that allowed non-Honeywell devices such as Modicon and Allen-Bradley PLCs to be interfaced to the TDC 2000 and a firmware enhancement to support sequence capability using SOPL (Sequence-Oriented Programming Language)—this was the first time a control-engineer-friendly language became available in a controller.

The TDC 2000 was introduced with the theme “A System You Can Start With, Live With, And Grow With.” The basic controller, now 30+ years old, is still running many refineries worldwide. It has not been withdrawn from sale. The hardware platform has been recreated, due to parts obsolescence among other issues, as the “Universal Controller” product.

3.2 The TDC 3000

The first generation of distributed control systems were process *control* systems rather than process *automation* systems. Other limitations included a lack of discrete-event handling capability and the use of two separate operator interfaces (one for the supervisory computer and another for the basic controllers). The TDC 3000 (see Fig. 3) was Honeywell’s next major DCS release and intended to address these limitations. The TDC 3000 subsumed the TDC 2000; multiple data hiways and distributed controllers and transmitters could be integrated within one TDC 3000 system. A “universal station” replaced the different operator stations. The TDC 3000 also included a node which tied all automation activities together—the Application Module (AM). The AM was an advanced, supervisory, and direct digital controller capable of spanning multiple low-level control products. A much faster (5 Mbps) fully redundant local area network for control, called the Local Control Network (LCN), was also included.

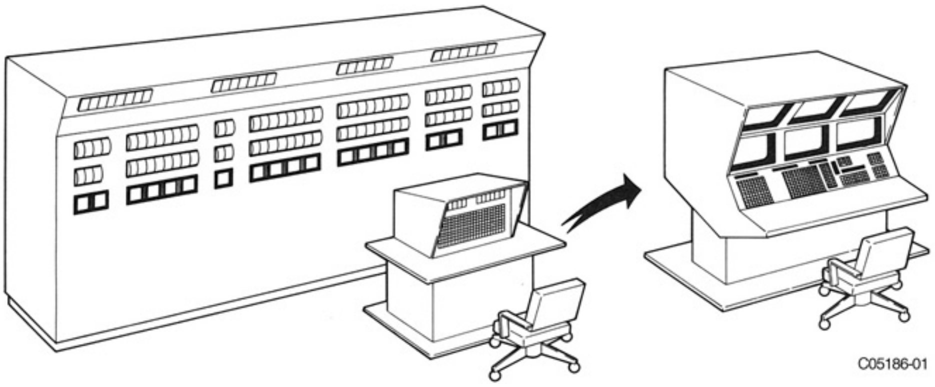

Fig. 2. TDC 2000 innovations included the replacement of the traditional instrument panel with analog displays (left) with a multi-CRT console (right). From (Dallimonti, 1985)

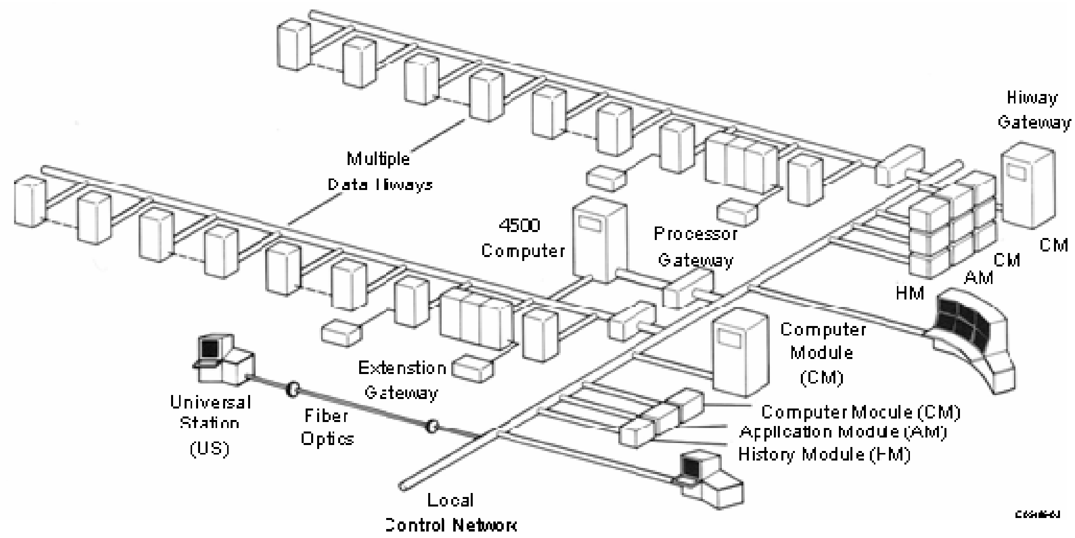

Fig. 3. The original system architecture for the Honeywell TDC 3000. The Local Control Network was introduced to integrate multiple TDC 2000 Data Hiways. The computer modules could be used for production scheduling and other information management tasks as well as for process control. From (Dallimonti, 1985)

A significant release of the TDC 3000 was R210, introduced in September, 1988. This brought in the Process Manager (PM) controller, the Universal Control Network (UCN), and the Network Interface Module (NIM). The PM controller featured dual redundant Motorola 68000 microprocessors and performed 160 loops/sec (with the R500 release of the High-Performance Process Manager with 68040 processors, the execution bandwidth increased to 800 loops/sec). The UCN was used to network PMs into the TDC system with the NIM serving as the LCN/UCN gateway.

With the TDC 3000, one automation system could be used to oversee and regulate the operation not just of a process but of an entire facility.

3.3 Recent Architectural Developments

More recent architectural, or architecturally relevant, developments in process automation include the following:

- Process control has historically been considered a continuous control application with occasional and limited need for discrete or event-based control, and specialized devices have generally been used. With the emergence of hybrid control applications, a need for controllers that integrate different time- and event-based control has arisen. Hybrid control in this context encompasses regulatory, discrete, batch, logic, and sequence control.
- Proprietary systems (such as the TDC) have given way to "open" systems, typically based on PCs running variations of Windows. PCs are not hosting closed-loop control for safety-critical applications, but they are now in widespread use for supervisory functions and advanced applications. Even for these purposes, extensions have been made to off-the-shelf systems, such as an ability to designate windows that cannot be occluded on the desktop.
- A related development is the popularity of fieldbuses—device-level digital communication networks that conform to established standards.

Several fieldbuses are now in widespread use, including Foundation Fieldbus, ControlNet, Profibus, and Modbus.

- In another related development, fieldbus technology has led to devices becoming more sophisticated. Sensor transmitters can have compression and scaling algorithms built in, and actuators can include processors on which control calculations can be executed. The control system has become even more distributed, with the potential for negative impact on overall system latency and jitter.
- With Moore's Law continuing its seemingly inexorable progress, more and more computing power has steadily become available at all levels of the automation architecture. Small multivariable model-predictive control (MPC) has migrated from level 3 (supervisory) to level 2 (regulatory) controllers, the difference being that the latter are embedded. This trend will continue, but we believe the emphasis will shift toward highly available MPC with easy-to-use interfaces and tools. Level 2 controllers are typically designed for hosting PIDs and their users are traditionally instrumentation technicians, unit operators, foremen, and supervisors. Architectural modification is needed before MPC and its supporting tools can be "natively" integrated into the embedded controller to further lower the knowledge required for implementing advanced control in closer-to-the-process controllers.
- Recent technology revolutions such as the World Wide Web are now finding their way into process automation systems. For example, process industry customers can subscribe to Honeywell's Loop Scout service (www.loopscout.com) and have process data automatically collected, communicated over the Web to a remote server, and analyzed by algorithms hosted on the server. Reports are then delivered to the customer via the Web (Fig. 4). The Web and the Internet are also enabling support for remote access and operation.

These and other recent developments are already making their way into commercial DCS offerings. For example, Honeywell's Experion PKS (www.experionpks.com) permits the integration of non-Honeywell process control and safety systems, interfaces with multiple fieldbus protocols, and uses Web technologies to provide a unified facilitywide view for local or distant staff.

Experion PKS also includes a control solution framework called the Control Execution Environment (CEE) which addresses hybrid control requirements. The CEE evolved from the marriage of process industry needs and emerging computer science theory—object oriented analysis and design. It was recognized that early DCSs forced users to think in terms of the control systems themselves and not of the process under control. Instead of thinking in "natural" terms of pumps, boilers, reactors and the like, process engineers were forced to model and design in terms of "points" and "parameters". The CEE approach was to apply object orientation to control system design so that users could design their automation configuration in the "natural paradigm," using function blocks organized in hierarchical control structures that mapped 1:1 to process elements and which could be templatized and reused for much faster and more reliable engineering.

In its latest version, CEE features a backplane-less design, using Fault-Tolerant Ethernet and a new I/O link as the basis for joining peer and subordinate devices. This packaging allows for much greater flexibility and mix of traditional and emerging IO devices, especially the various fieldbus networks and devices (Foundation, ProfiBus, HART, et al.).

The flexible, graphically oriented design paradigm that is exemplified by CEE is not the only approach that is needed for process control. Especially for larger-scale applications—e.g., model-predictive control for a unit or an optimization application—an ability to design and analyze at more aggregated levels is also essential. To this end, in the next section we discuss advanced application infrastructure as an emerging technology.

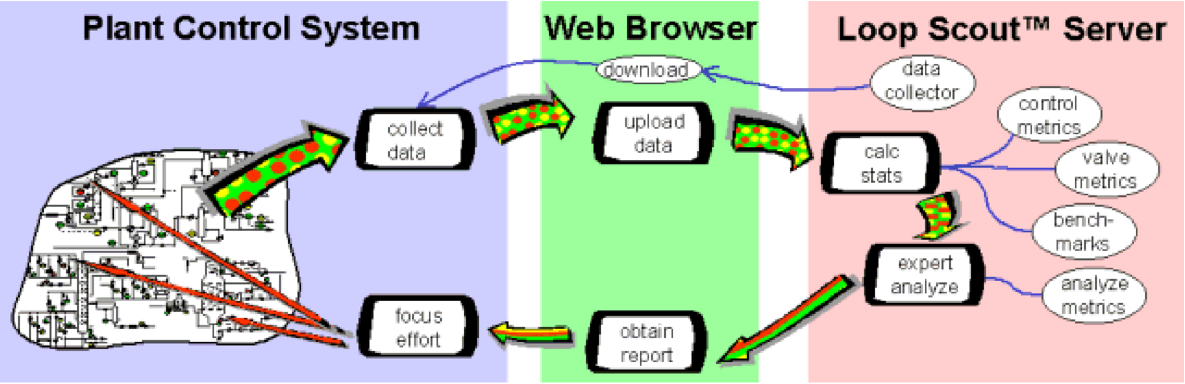

Fig. 4. Information flow for LoopScout^{TM} (www.loopscout.com).

4. EMERGING TECHNOLOGIES FOR PROCESS AUTOMATION

4.1 Wireless

The replacement of wired communication with wireless may not seem a transformational change, but it is. Wireless enables substantial reduction in installation cost, it enables untethered mobility for humans and machines, and it enables measurement and monitoring of a quality and scale that could not previously have been envisioned.

The cost of wiring in an industrial plant has been estimated at up to $100 per foot and up to $2000 per foot for specialized applications (DOE, 2002). As noted above, the cost of equipment such as a sensor is generally substantially less than the wiring cost associated with installation. With broad-based adoption of wireless savings in the millions of dollars are likely to be realized.

Wireless is an architectural innovation that begets others. Today access to the control system is obtained from the control room or offices, through fixed displays. Already, portable wireless terminals are available in the market and being used in process plants. These devices provide limited functionality compared to what is available in the control room today, but we can envision a future where operators and other staff can obtain much the same information that is available to them in the control room virtually anywhere in the plant—for example, at the location of a potential problem. In effect, the control room will likely become distributed.

Although we measure the variables that we need to measure for effective operation of a plant, there are many gaps in our measurements that limit performance and reliability. With sensors becoming increasingly miniaturized and wireless promising an order-of-magnitude-plus reduction in installation cost, we can foresee much more use of sensing. Wireless may be a harbinger of a "pervasive sensing" paradigm in process automation. Sensors could also be installed on a temporary basis for asset management—e.g., when there is an early indication of degradation, to defer replacement until necessary with minimum down time.

Wireless technology has to progress significantly before these visions can be realized. Two challenges in particular are critical to overcome for process industry applications: power management and reliability. With thousands of potential wireless devices, battery lifetimes on the order of a year or so will require full-time staff just for battery replacement. Better energy storage and harvesting technologies as well as more intelligent power usage approaches are needed. In the foreseeable future, wireless devices may often be line-powered—communication, not power supply, may be wireless. The reliability problem would also be ameliorated in this case, since higher-power transmission would exact less operational cost.

4.2 Intelligent Network Devices

Automation architectures are often structurally complex and hierarchical because of the variety of different protocols and communication media used for different functions. Integration requires extensive use of a variety of bridges and gateways. Recently, a new class of network devices have appeared that can directly connect to multiple networks and serve as unified intelligent gateways (Tridium, 2003). Ports for multiple types of network connectors and protocol conversion between a supervisory IP (Internet protocol) network and a variety of control networks including Modbus, Cnet, LON, and BACnet can be integrated within one device.

Network-connected devices now also increasingly feature embedded Web servers. Through any Internet connection, these devices can serve up Web pages including graphics, parameter values, and configuration screens. Lower-level sensor and control equipment that is connected to these devices can then be directly configured and monitored from a browser anywhere through a secure connection.

This technology is first making its mark in building management systems (Fig. 5), but it is relevant (with appropriate caveats or modifications) for industrial automation as well.

We note an intriguing research challenge in this context. A representation formalism needs to be developed for the process industries that can capture the variety and complexity of process equipment in a structured way that explicitly records semantics, relationships, and dependencies.

4.3 Service-Oriented Architecture

Today's automation systems deliver much of their functionality through software programs: monitoring, estimation, control, and optimization algorithms; visualization, trending, and other operator aids; integration bridges with business and supplier databases; etc. We often refer to such software as "applications," but this term has connotations—packaged, stand-alone, purposefully obtained—that can be misleading as new software architectures are developed and adopted.

An exciting example of a new software architecture methodology is service-oriented architecture (SOA) (Reekie and McAdam, in preparation). SOA is founded on the provision and consumption of "services," which are software programs in a distributed computing environment. Applications become much more loosely bound to one another through exposed service contracts as opposed to the

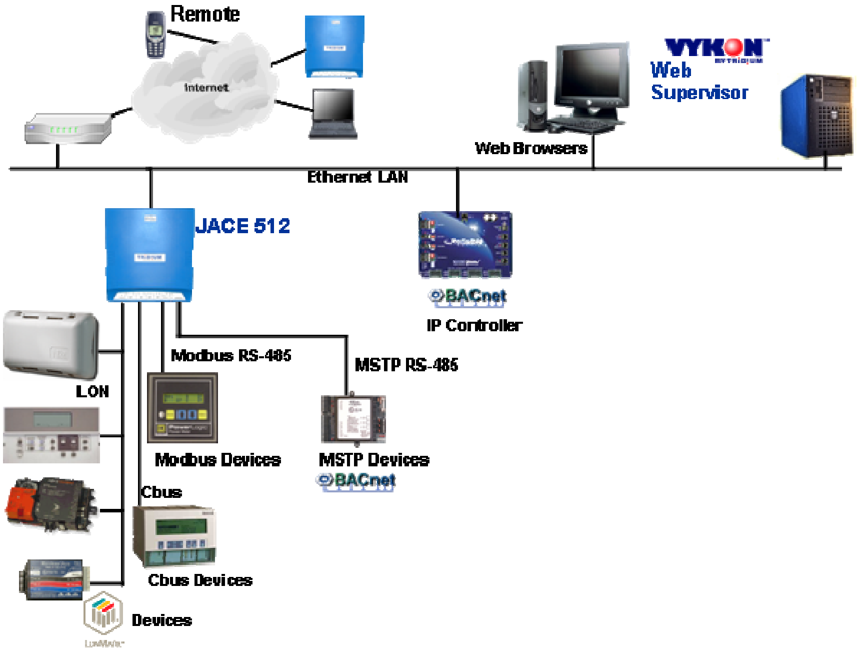

Fig. 5. A new architecture for building management systems, with intelligent network devices featuring embedded Web servers and multiprotocol capabilities (www.tridium.com).

prior model of application programming interfaces (APIs). These programs can be resident on a remote, connected site (typically via the Web). They are autonomous in the sense that they can serve useful functions on their own, yet they can be composed with other services when needed; their functions and interfaces are well-defined in an explicit modeling language (e.g., some XML variant or extension) with the result that other programs can automatically reason about them and compose them. SOAs also provide automated discovery and publishing, with the result that new services can be developed and integrated at any time with as much (or as little) human supervision as desired. In principle at least, a university team could develop a new monitoring or visualization tool and "publish" it on its Web site. It could be discovered by the automation system of a plant which would recognize that an improved monitoring service is needed. A trial version could be downloaded for offline testing. Once verified, it could be brought online. Commercial service providers would be able to provide a pay-per-use service which could be automatically negotiated and engaged. Plantwide models could automatically be composed from component models developed by different vendors, with tuning to the actual plant done through another service. Thus, among other benefits, SOAs open up new possibilities for business delivery and collaboration models.

The main challenge in the broad-based adoption of SOA is in how loosely coupled, widely distributed services can be integrated in an automation system in a way that is consistent with architectural qualities such as reliability, availability, manageability, migration/upgrade, and security.

4.4 Infrastructure for Advanced Applications

The automation system is being envisioned as the control and decision support centre for just-in-time manufacturing. It is asked to handle not only basic and advanced control, economic optimization, production/maintenance scheduling, and long-term planning, but also asset management, decision support, best practices implementation, and overall business agility improvement. The trend is also driven by the desire for broader participation from users, vendors, consultants, and third parties in various automation activities and practices.

An aspect of system architecture that relates to the ability of the automation system to serve these functions and stakeholders is the infrastructure that the architecture provides for developing, deploying, and maintaining advanced software applications.

The attributes required for an advanced application infrastructure include (a partial list):

- support for resources and services that can handle a mix of "hard" real-time, "soft" real-time and non-real-time applications;
- efficient execution of a mix of continuous, discrete, and transaction-based applications;
- a suitable namespace and organizational scheme;
- an ability to process complex data in system components and services, including communication schemes, data integrity, and presentations and timely updating of complex data;
- support for version control, application deployment, update, and migration;
- ability to isolate an application crash from the rest of the system—particularly needed for customer-created applications;
- support for application "plug & play" and componentization.

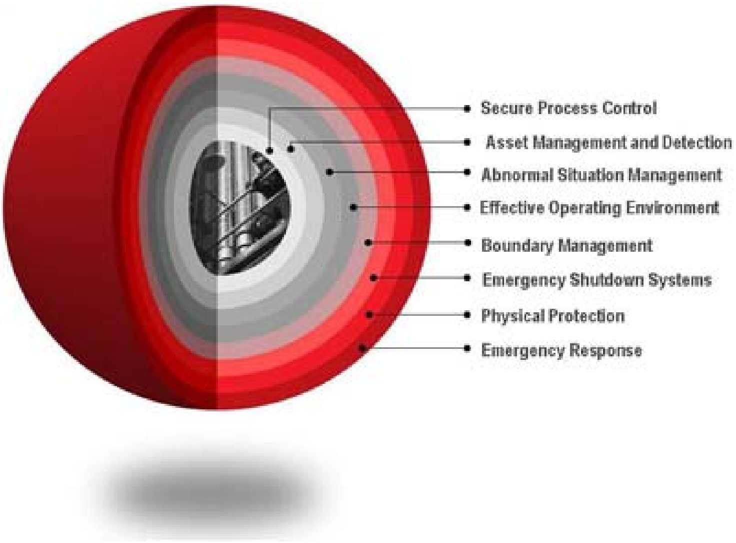
Secure Process Control
Asset Management and Detection
Abnormal Situation Management
Effective Operating Environment
Boundary Management
Emergency Shutdown Systems
Physical Protection
Emergency Response

Fig. 6. A layered protection framework for integrated process safety. From (Honeywell, 2005)

See Horn et al. (2005) for a recent application infrastructure development, Uniformance Real Time (URT), that addresses several such issues and Havlena and Lu (2005) for examples of how the development and deployment of advanced applications can benefit from such infrastructure.

5. INFORMATION TECHNOLOGIES AND PROCESS AUTOMATION

Progress in automation architecture since the advent of computer-based control has steadily loosened constraints of collocation, bandwidth, integration, and access. But it seems only a slight overstatement to say that recent technology developments promise to completely remove such constraints. Trends in system architecture portend connectivity on demand, boundaryless systems, and global information access. This is not to say that future process automation systems will not impose any restrictions on human-automation-process interactions, but that the restrictions may be based on functional and operational requirements, not derived from technological limitations.

In our prognostications we must not forget that we are concerned with process automation, not office automation. The "CTQs" we noted earlier cannot be ignored. Reliability or real-time responsiveness cannot be compromised when we are considering process-connected equipment. Installation cost concerns will continue to often trump technology innovation. Whether through "loops-per-operator" or (more likely) some other metric, staff size will remain a point of scrutiny. The trends we have highlighted did not arise in the process control space; as with microprocessors, CRTs, and communication networks they are being borrowed and extended from more general information technology developments.

The security challenge is worth emphasizing. With process automation evolving toward open, integrated, wireless, nonlocalized system architectures, there are potentially many more points of access and hence vulnerability. Hackers, phishers, spammers, not to mention terrormongers must be guarded against. But the intersection of the physical and the IT world that is represented by process automation implies that cyber and physical security must be coordinated and that safety and security must be jointly considered. Thus, there is no single solution or silver bullet. Multiple protections and a holistic perspective are necessary. In this context, Fig. 6 illustrates a layered framework for process safety and security, with different levels of protection and response available to manage a variety of problems. Today's automation systems already have footprints that extend into cyberspace beyond the control system per se and into physical space beyond the plant perimeter (e.g., wireless signals). A sophisticated, multifaceted approach is recommended, and will, with appropriate extensions, be essential as economics and performance considerations drive us toward even more open and globally connected architectures.

6. CONCLUSIONS

Research in process control has by and large been an algorithmic enterprise. Variations of PID, automatic tuning, loop shaping, model-predictive control, filtering and estimation, inferential sensing, statistical process monitoring, sensor validation... developments in these areas are the marks and milestones of progress. It is not just in the research literature where these contributions in algorithms and theory have made an impact; industrial processes operate more reliably, more efficiently, and more productively as a result of this knowledge base.

It is easy to overlook the role of automation system architecture in the advancement of process control practice. Enhancements in computation and memory, communication networks, operator interfaces and interaction modalities, sensing and actuation infrastructure, software technologies, and design, development, and deployment capabilities have been necessary preconditions for deriving operational benefits from research results.

Developments in information technologies continue apace. Indeed, we have claimed in this paper that the convergence of new technologies today is reminiscent of an earlier convergence, one from three decades ago and that led to the development of the first DCS with all the attendant benefits of that revolutionary advance. This assertion raises interesting questions, in particular: What new research directions in process control will be motivated by these architectural innovations and what dramatic practical improvements can be envisioned? These questions will be answered in course, but by anticipating the answers the research community can help shrink the research/practice gap and expedite the process of achieving economic impact.

ACKNOWLEDGEMENTS

We thank R. McAdam for contributions to the material on service-oriented architecture.

REFERENCES

Dallimonti, R (1985). The development of TDC 2000. *Scientific Honeywell*, **vol 6**, no. 4, pp. 23-28.

DOE (2002). Industrial Wireless Technology for the 21st Century. Office of Energy Efficiency and Renewable Energy, U.S. Department of Energy. Available online at http://www.eere.energy.gov/industry/sensors_automation/pdfs/wireless_technology.pdf

Havlena, V. and J. Lu (2005). A distributed automation framework for plant-wide control, optimisation, scheduling and planning. In *Selected Plenaries, Milestones and Surveys, 16th IFAC World Congress*, P. Horacek et al. (ed.), pp. 80-95.

Honeywell (2005). Integrated safety. Available online at http://hpsweb.honeywell.com/Cultures-/en-US/Services/IndustrialSecurity/Cyber-Security/default.htm.

Horn, B. et al. (2005). Platform for advanced control applications. *Proc. 16th IFAC World Congress*, Prague.

Moore, J.F. (2003). Creating profit with computers: my life as CEO of Bonner & Moore Associates. *Annals of the History of Computing*, **vol. 25**, no. 3, July-Sept. 2003 pp.30 - 47.

Przybylski, F. (1989). Industrial control: a tutorial. *Scientific Honeyweller*, vol. 10, no. 1, pp. 6-18.

Reekie, J. and R. McAdam (in preparation). *A Software Architecture Primer*.

Tridium (2003). *Unifying Automation Systems with a Web-enabled Software Platform: The Need for an Automation Framework.* White paper. Available online at http://www.tridium.com/library/whitepaper/UnifyingAutomationSystemsWithWebEnabledPlatformWP.pdf.