SEPO COLLABORATIONS

Inside Knowledge for MITRE's Systems Engineering Community

Volume 2, Issue 2

Prototyping @ MITRE

The Power of Prototyping

by Judith Clapp

THIS ISSUE OF COLLABORATIONS IS A COLLECTION OF ARTICLES ABOUT PROTOTYPING AT MITRE. We have been using prototyping in many ways.

- Helping users and developers understand and verify requirements, especially new operational concepts, and system-to-system and user interfaces.
- Understanding and verifying design decisions such as algorithms for processing new sources of data; predicting system throughput, load, and response time; and verifying the interface to new hardware and other equipment.
- Determining the maturity of disruptive technology and technical products and their suitability to system requirements and design through hands-on experience that connects the researchers to the users.
- Performing experimentation to learn where there are fieldable solutions.

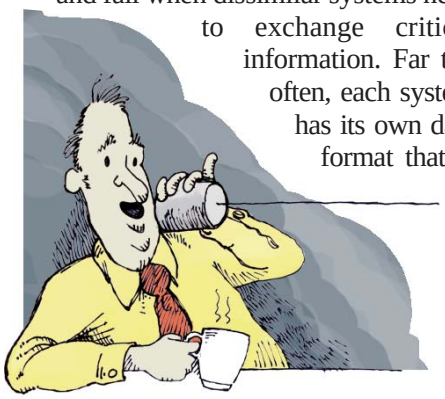
The articles in this newsletter and our work at MITRE illustrate that there really is no single, simple definition of a prototype. The common thread is that prototypes delve deeply into a small aspect of a total system or system of systems and develop a capability rapidly. The effort to develop a prototype is usually small proportional to that for developing the system with which it will interact or will influence. In addition to examples of prototypes, we have included articles on deciding what to prototype, a development environment and tools for rapidly producing software for prototypes, and options for transferring prototypes from MITRE to other organizations for continued support and further uses.

Making Net-Centric Warfare Real Today

by Rich Byrne

Recent successes in Afghanistan and Iraq introduced the world to a new military paradigm. Joint operations, information superiority, and bringing to bear the power of the full military network to every fight are touted as evidence that we are achieving net-centric warfare. However, in reality we are just in the infancy of this revolutionary shift in capabilities.

In the battlefield what we normally have are islands of stovepiped communications that can only pass information between the same types of equipment and fail when dissimilar systems need to exchange critical information. Far too often, each system has its own data format that is complex and non-interoperable with other systems without slow, and error-prone, human-intensive translations. While we wait for massive improvements in our communication networks and reductions in our stovepiped systems, we can truly benefit from the promises of machine-to-machine net-centric warfare today.



Using rapid prototyping we can simplify our focus to just the most important information that needs to be shared. We can teach each system how to translate just a few words from its "native" language into a common "esperanto" language. The cost and complexity can be so greatly reduced that this can be accomplished today in weeks not years.

After an analysis of what goes on in modern battles, we have found that "what," "where," and "when" constitute the most valuable information for an amazingly large number of the most critical missions that need to exchange information across many systems. "What" tells us if this is a friendly or hostile force, a target to be killed, or a survivor to be rescued. "Where" has become synonymous with military accuracy of precision coordinates that guide munitions through windows or navigate tanks through zero-visibility sandstorms. "When" is becoming increasingly important as we dramatically shrink the sensor-to-shooter timeline for time-sensitive-targeting missions.

continued on page 2

continued from page 1

The Air Force and MITRE have undertaken several initiatives to rapidly deploy just such an information strategy to fielded systems. The software required is modest indeed (a few hundred to a few thousand lines of code). It can be readily added to existing fielded equipment (no new hardware), and it is compact enough to run across even the oldest radio links and networks being used (no need to wait for new systems to become net centric). Machine-to-machine information exchanges are then easily automated (no more slow, error-prone human translations) with humans only in the loop for the key decision making steps. For example, friendly locations can be shared from the set of blue force tracking systems to systems that previously couldn't receive it—reducing the probability of fratricide.

This strategy can be applied to intelligence, surveillance, and reconnaissance (ISR) information flows, such as those from unmanned air vehicles (UAVs) to enable the warfighter to get ISR information "on demand." The AF/MITRE work has drawn inquiries from dozens more systems regarding how they can add their universal What, Where, and When capability. Longer term activity is also needed to realize the full potential of net-centric warfare. Robustness, information management, and information assurance are examples of areas needing more attention. Nevertheless, the Department of Defense should seize the opportunity today to rapidly tie systems together using the What, Where, and When approach.

Rapid Prototype Joint Deployable Monitor (JDM) Meets the JFC's Needs

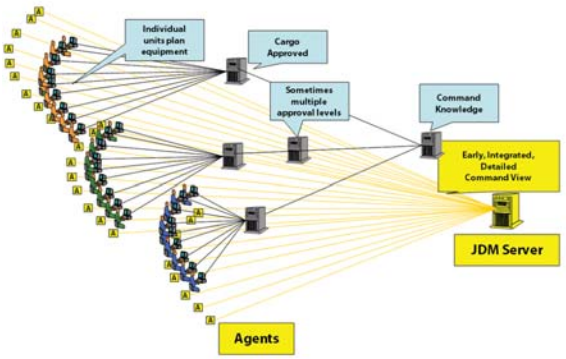
by Robert Pancotti & Andrew Gregorowicz

Current decision support tools and information systems that assist commanders in moving equipment and troops for military deployments involve the use of a collection of isolated systems. Each has its own parts of the data needed. They do not provide the joint force commander (JFC) with a coherent and comprehensive capability with which to plan, analyze, and execute the joint deployment process. New capabilities are required to allow information from all Services to be shared in an integrated picture of tactical-level deployment data.

The JDM prototype gathers common cargo deployment data from various Service deployment systems. JDM exposes "raw" deployment data (data independent of the tools used to visualize it) for use by commanders. Management and decision making at all levels are enhanced by early and continuous access to the same data, at the same time, and at the highest level of detail possible. This is accomplished in the JDM prototype by using Web Services technology and agents that connect to Service deployment databases and retrieve data in an XML format common across all Services. The agents publish deployment data through a web service to a central J2EE-based server. This server can then be accessed by a web service or web interface to retrieve information. To build this system rapidly, the team relied heavily on open source technologies such as JBoss, Apache Axis, and Struts.

The JDM improves high-level visibility of the deployment process, significantly reduces the time to view joint deployment data and execute a plan, integrates divergent stovepiped systems without modifying the systems or existing deployment planning processes, and provides a joint enterprise service for deployment information.

For more information on JDM, contact Robert Pancotti, rlp@mitre.org, 781-377-1159; or Andrew Gregorowicz, andrewg@mitre.org, 781-271-8228.



JDM uses Web Services technology and agents to connect to and retrieve data in an XML format common across all Services.

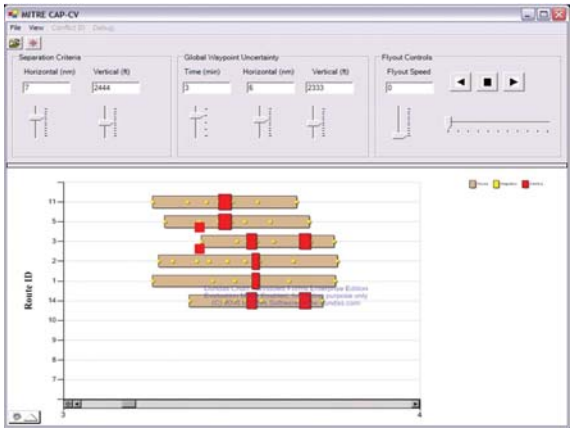
Rapid Prototyping Lessons Learned *by Rich Byrne*

- Don't compete with programs; augment their efforts by addressing gaps, reducing risks, exploring "what ifs."
- Make sure the prototype satisfies a real need.
- Build and maintain a strong relationship with the user.
- Choose a user who is open to change.
- Gain user advocacy early.
- Build and test in the user environment or keep going back to the user.
- Plan to make iterations frequently.
- Leverage what is available today in the field.
- Keep the scope small so the value is much greater than the effort.
- Make sure the prototype is better than what the user has now.
- Keep the technical effort trivial.
- Remember that building is a small part of the effort compared to time for demonstrations, certification, and transition.
- Build a thin layer on top of the existing systems and equipment.
- Make sure it does not disrupt what is already operating by assuring it can be removed if it fails or is not acceptable.
- Plan for transition so you do not have to maintain it.
- Consider the role of contractor as contributor to the prototype or its recipient.
- Ensure someone produces documentation and user manuals that are part of the transition package.

Commanding Air Power—Conflict Visualization (CAP-CV) Rapid Prototyping Initiative

by Michael Dinsmore

MITRE and Electronic Systems Center (ESC) are conducting a rapid prototyping initiative called "Commanding Air Power–Conflict Visualization" (CAP-CV) to provide the warfighter with automatic air battle plan conflict identification and visualization. The initiative allows a user to define a "conflict" (e.g., what aircraft have the potential of colliding in a given place and time), and the system shows where those types of situations can happen. Many users, including those at the Air and Space Operations Center (AOC) and at the unit level, could benefit greatly from a focused, guided visualization of conflicts. The capability could also be used as an interface for a commander to visualize the overall air tasking order (ATO) at a detailed route level.



The CAP-CV Interface allows users to "dial in" specific parameters to highlight potential conflicts.

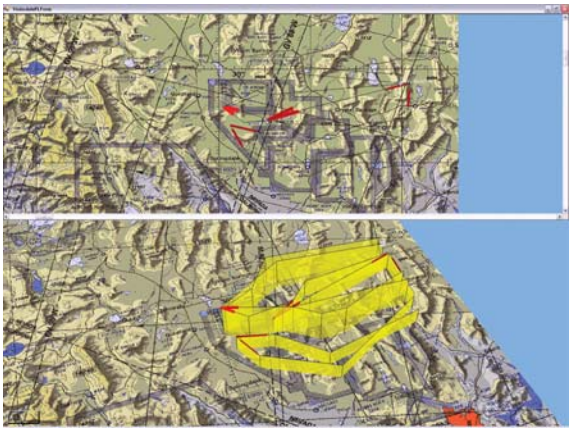
The CAP-CV interface leverages many of the visualization techniques being developed in the Center for Advanced Aviation System Development (CAASD) for commercial airspace management. The prototype was designed with an open, modular design to facilitate experimentation. EXtensible Markup Language (XML) definitions separate the various modules and enable interoperability. This approach has been successfully used to experiment with different visualization toolkits, different algorithms, and different data sources.

> CAP-CV provides the warfighter with automatic air battle plan conflict identification and visualization

The CAP-CV team focused on the user interface and services manager component, working with Human Factors Engineers and Subject Matter Experts to quickly iterate the visual user interface. Slider bars are presented, which allow the user to "dial-in" the parameters for what they consider to be a conflict. Each user may have different situations, so this enables flexibility and serves more as a decision aid rather than an automatic decision maker.

The interface was prototyped using the C# language in Microsoft's new .NET environment. This prototyping environment has shown itself to be very flexible and enables concepts to be instantiated quickly. For map displays and three dimensional (3-D) visualization components, CAP-CV uses several commercial toolkits, such as the upcoming Commercial Joint Mapping Tool Kit (C/JMTK) and the Satellite Tool Kit (STK) for the visualization module for a different look and different visual options.

CAP-CV can access information from various XML sources. Mission Planning has defined a Common Route Definition (CRD) XML schema to enable transfer of route information between systems. By incorporating this CRD XML schema, CAP-CV can import flight routes from many different sources, including the Portable Flight Planning System (PFPS), the Joint Mission Planning System (JMPS), and other CRD producers.



The CAP-CV visualization module uses commercial toolkits for tailoring different map views

For more information on CAP-CV, contact Michael Dinsmore, dinsmore@mitre.org, 781-271-5234; or Carol Hubbard, hubbard@mitre.org, 781-271-5406.

The MITRE Corporation's Virtual Room (VRoom) Provides an Effective Prototype Development Environment

by Dave deMoulpied

In a crisis, whether military or civilian, an emergency management decision maker wants as much information as possible and all of it accessible at once. The information usually comes from multiple sources in multiple forms. It may include maps, tables of data, and other information updated in real time. A typical PC-based desktop display presents overlapping windows, obscuring much of the information.

The MITRE Corporation's VRoom prototype is one of the first desktop systems to use three-dimensional graphics to present a room-based environment for viewing multiple displays. Its technology is based on the premise that a three-dimensional environment is more effective than a two-dimensional computer desktop for people trying to filter and absorb large quantities of information from a variety of sources. The VRoom creates a flexible three-dimensional virtual room in which a computer user can visualize a variety of applications and interact with them by placing the applications on the "walls" and floor of a simulated room. The viewers can have any number of walls with their data feeds and can pan around the room to select which windows to view without the clutter of a typical desktop display.

With VRoom's architecture the user can easily add new data sources to its environment, seamlessly integrate custom 3-D visualization components, and configure intelligent software agents to filter information, provide alerts, or perform domain-specific jobs. Its most comprehensive visual system, referred to as the Sand Table, translates a two-dimensional map into a three-dimensional viewing tool.

The VRoom framework allows for real-time interaction with local and remote multi-platform applications, providing remote users with a sense of context essential to collaborative environments and distributed command centers. VRoom provides an effective environment for training on new equipment and for prototyping new configurations of equipment. The VRoom framework has already become the basis for other prototypes presenting many kinds of information. The prototype operates on a consumer-level PC with a 3-D graphics card.

For more information on the VRoom, contact Dave deMoulpied, ddemoulp@mitre.org, 781-271-7876.

The VRoom offers an easily organized, 3-D environment with intuitive controls to help people process large amounts of information.

Multi-User Detection (MUD) Technique Processes Multiple Simultaneous Signals

by James Dunyak

The need to move more data in less time via wireless links has resulted in an increasingly crowded radio-frequency spectrum, particularly in military settings. Co-channel interference has become the limiting factor in the performance of communication, navigation, and sensor systems; achieving optimal performance demands new interference cancellation methods. MITRE, in close partnership with the Air Force Research Laboratory and contractors, has designed a ground-breaking prototype to process multiple, simultaneous signals.

Using a Multi-User Detection (MUD) technique developed under a MITRE-Sponsored Research (MSR) prototyping program, direct sequence spread-spectrum communications systems can increase the total number of simultaneous users. The joint MITRE, government, and contractor team's accomplishment is notable because past attempts at solving this problem resulted in algorithms that were too computationally complex and impractical to implement.

The technology developed during this prototyping shows significant potential in both the defense and commercial sectors, resulting in a patent application. Most significantly, a U.S. government program has already adopted the technology for integration into a larger system and tested it in an operational environment, where it will provide a critical capability for the war on terrorism. During a recent review, the Air Force Science Advisory Board singled out this system as "relevant to Air Force needs," "impressive," and "world class."

For more information on MUD, contact James Dunyak, jdunyak@mitre.org, 781-271-3555.

The MITRE Open Source Development Environment

by Tim Taylor

The MITRE Open Source Development Environment (MOSDEV) is an emerging capability being developed by D520, eServices. The objective is to create a suite of tools and a development process supported by as many automated open source tools as possible to help our development teams realize the following goals:

- Reduced cost to produce services.
- Improved quality of the software produced.
- An agile organization capable of responding to changing program needs.

Productivity goes up when the developer is familiar with the tools and libraries being used on the project. Using MOSDEV, the time that would ordinarily be spent "ramping up" and becoming proficient at the latest and greatest technologies is instead spent working the real problem. In addition, the potential for cross-project reuse of components is increased when all projects are using the same tools, programming language, and libraries.

Software quality is improved when frequent tests are run and rapid feedback on the current integration status of the project is made available to the developers. The Extreme Programming methodology emphasizes "continuous integration" to help avoid the chaos that accompanies traditional "milestone" integration events.

When all the projects are using common tools, libraries, and processes, it is possible then to more easily move people from one project to another as funding profiles and project requirements demand. The developers being added to a new project now just have to "ramp up" on the domain knowledge specific to the new project without having to learn the particular suite of tools that this particular project chose to use because the whole section is using MOSDEV.

Development Lines

In looking at the types of projects in D520, we were able to classify the projects into two broad categories. The first class was one in which we had full control over the deployed solution. In other words, the customer desired a "turn-key" solution and didn't care how it was packaged or what it ran on as long as it solved their problem. As it turns out, this is where the majority of our projects tend to lie.

The second class was one in which the customer dictated the environment to which the final product would be deployed. Projects that fell into this category were constrained by the platforms, application servers, and approved third-party libraries that they were allowed to use in the product.

In looking at the unique requirements each of these categories presented, it was clear that a single suite of tools would not be able to satisfy the needs of both. Instead, we have created what we call development lines that are tailored to a specific customer category. Currently we have one active development line, our mainstream line, which is used by most of our projects. We are also considering additional lines, such as an internal line to be used for ISIS-related work. The development lines have as much in common as possible given the requirements of the target customer class. The development lines are kept stable for a period of time determined by the customer class. For example, our mainstream line is reviewed every six months to determine if new versions of current components should be added, or if a component should be retired in favor of something else. By keeping the contents of the lines stable (only security-related updates are done to the lines between reviews) the developers learn about the quirks of a given version, and how to work around them.

Continuous Integration

The other significant aspect of the MOSDEV is the continuous integration (CI) that is performed on projects. This is still a work in progress, but the objective is to be able to provide rapid feedback to the development teams on the current state of the project. Within 15 minutes of checking code into the Concurrent Version System (CVS), one of our continuous integration machines will have checked out the modified code, built it, run any tests currently defined, validated the code against project-team accepted coding standards, built the API documentation, and any other project-related documents, and published the results to the projects web site. A problem with the build process results in an email alert sent to the project leader, and all developers that contributed changes that went into that particular build.

The fact that we have standardized development baselines makes it possible to have a CI machine configured for each baseline that can be used for building and testing any projects using that baseline.

Mainstream Development Line Contents

The cornerstone of our mainstream development line is the Fedora Core 1 linux distribution. On top of this, we run the JBoss J2EE application server. We target the Java 1.4.2_03 JVM and J2EE 1.3.1. Some of the Open Source tools in the suite include Struts, Castor, Axis, Log4J, JSTL, Xerces, Xalan, Velocity, and Lucene. For project build and management, we use Maven to drive the build process, which also leverages Ant. XDoclet is used to create the many deployment descriptors required by J2EE.

For more information, contact Tim Taylor, ttaylor@mitre.org, 781-271-2099.

Blue Force Tracking

by Bruce T. Robinson

In January 2002, the Army component of U.S. Central Command asked the Army's Chief Information Officer to provide an information management assessment of his network and to assess potential improvements to the command center in Kuwait. As a result, a task force of military personnel and contractors, including a MITRE representative, was dispatched to conduct the assessment. One recommendation was to bring a Blue Force Tracking (BFT) capability into the Southwest Asia theater of operations.

BFT is a system that allows a commander to track his forces on the battlefield, and provide situational awareness and Command and Control. At the time of the task force's evaluation, the Army's 4th Infantry Division was using a system, Force XXI Battle Command Brigade and Below (FBCB2), for training in Ft. Hood, Texas. A satellite-controlled variant of FBCB2 had also been used to prevent inadvertent border crossings in the Balkans. FBCB2, however, had yet to be proven in combat.

As a result of the task force's recommendation, the Army Staff in October 2002 decided to provide ARCENT and CENTCOM with the new BFT capability by February 2003, in preparation for Operation Enduring/Iraqi Freedom. The FBCB2 Program Manager at Ft. Monmouth, N.J., was given the lead for this immense five-month effort. MITRE, along with a team of other contractors, provided engineering, logistics, installation, training, and field service support.

For BFT to work in Kuwait, the operational capability of the FBCB2 framework used in the Balkans had to be enhanced. It needed greater bandwidth, and its use had to be expanded from trucks to helicopters, Bradley Fighting Vehicles, and Abrams M-1 Tanks.

Among the first tasks was to find a satellite system that could provide the necessary geographic coverage. The Ku band coverage areas in Europe for BDI proved to be inadequate for Southwest Asia. L band was the communications of choice. MITRE developed a micro-router that allowed for seamless transition between Ku-band and L-band communications. Also, redundant satellite services had to be procured, and a satellite hub facility established in Kuwait. The next step was building a Command Center in Camp Doha, Kuwait, in which the BFT Information could be displayed. To do this, MITRE partnered with Army scientists and engineers at Ft. Monmouth, the Pentagon, and in the field, as well as Northrop Grumman and a number of other contractors. Despite seemingly overwhelming challenges, in the five-month period from October 2002 to March 2003, BFT was installed and staff trained concurrently in six different countries and over 20 continental states.

The situational awareness provided by BFT reduced many fold the "fog of war"—the confusion on the battlefield that leads to casualties. Secretary Donald Rumsfeld cited BFT as one of the "transformational" technologies that proved critical during the Iraqi conflict.

For more information, contact Bruce T. Robinson, brobinso@mitre.org, 732-578-6434.

Software Prototyping and Commercial Technology Transfer

by Gerard Eldering

The Technology Transfer Office (TTO) facilitates the transfer of MITRE's technology to the commercial sector. When MITRE develops software prototypes during the course of work and research, they are provided to sponsors and their contractors and eventually are transferred to government use. MITRE also transfers prototypes to commercial companies where the prototype can become a commercial product offered to industry, the government, and the public as commercially supported software.

What are the basics of software and technology transfer at MITRE? The MITRE Corporation "owns" the software we write, and generally the U.S. Government has an automatic, free license to use it. We also have the right to license out the software for other uses, but we do not act as a software company and therefore do not support the software. To use MITRE developed software, our sponsors must either select a contactor to support the software or be able to purchase support from the commercial marketplace.

The Technology Transfer process has two main steps:

1. Protecting the intellectual property (a legal name for new software or inventions). In general, software can be protected with a simple copyright statement or a more complex patent. The TTO and Legal office can help apply the right process.
2. Licensing the software. Software can be licensed through broad Open Source licenses that allow free downloads, Direct Licenses to specific companies, or through more complex Consortia Arrangements. Open Source licenses fit well with MITRE's culture, but the results of the release (i.e., how widely it is adopted and used) are hard to track. With a direct license, the software is distributed more narrowly, but MITRE knows what happens to it and can get appropriate credit or royalties. Generally Direct Licenses work best when the licensing company has to make a big investment in the software to get it ready for commercial sales.

MITRE has had a lot of experience with transferring software prototypes over the past several years.

MITRE has had a lot of experience with transferring software prototypes over the past several years. CAASD's Terminal Area Route Generation and Evaluation Tools (TARGETS) prototype has been licensed to numerous airlines and one commercial software company. The software is now widely used, and MITRE has to deal with issues such as version control, training, and support. The Air Force Center's Virtual Room, or VRoom, prototype has been distributed to several sponsors and licensed to three government contractors. Several foreign companies have asked to license VRoom, and MITRE now has to get appropriate export licenses. Each software licensing case tends to raise numerous issues for the TTO to work through.

Transferring software prototypes to the commercial sector is an important activity at MITRE. It results in software that is available, supportable, and affordable for our sponsors and industry. Staff, working with the TTO and their Chief Engineer, should carefully ensure that MITRE's intellectual property is protected and that the prototype is successfully moved to the commercial sector.

For more information on Technology Transfer, contact Gerard Eldering, eldering@mitre.org, 703-983-7132.

Software Internationalization and Localization

by Dave Anderson

As software has become an international commodity, the need for and value of internationalization, or I18N (which stands for 18 letters between the first and last letters of "internationalization"), and localization, or L10N, have grown. I18N is the process that abstracts locale-specific properties of data and presentation into modular components, thus making it much easier to then support a variety of specific locales, i.e., L10N.

- I18N of data includes supporting the representation application data (e.g., text, time, money) in a variety of internal formats, or mapping various external representations to a single internal format capable of representing the required range (e.g., Unicode).
- I18N of presentation includes enabling the replacement of text strings in the user interface as well as changing how the data and application are presented. For example, Arabic text is presented left to right, European numbers are presented with periods and commas reversed from the American style, and many Asian countries use calendars different than our Julian calendar.

I18N and L10N are particularly needed in MITRE's support of coalition work such as Translingual Instant Messaging (TrIM) and exploitation of foreign-language data with tools like Geospatial News On Demand Environment (GeoNODE). In response, tools have evolved and standardized to support the process. User interfaces now support many user locales with customized presentations, often providing mechanisms to manage user interface (UI) text strings as loadable resources, and present internationalized data in the style of the user's locale. Unicode was developed to enable text from the widest set of locales to be mapped into one common representation. Databases now support text fields in Unicode. Integrated Development Environments (IDEs), and even programming languages, have evolved to provide features to make it easy to write internationalized software and localize the result.

I18N is an aspect of software development that should take only a nominal amount of effort if it's identified as a requirement up front. It has a limited impact on the choice of tools, because most modern tools already support I18N. Still, it is incumbent on the developers to exploit these features when producing an internationalized application. This cannot be easily outsourced, although specialty service firms can provide L10N at reasonable costs.

For more information on software internationalization, contact Dave Anderson, dca@mitre.org, 781-271-2594; or Galen Williamson, gwilliam@mitre.org, 781-271-3263.

Who We Are

The MITRE Systems Engineering Process Office (SEPO) is a nexus for systems engineering information and activity at MITRE. Our team brings together useful systems engineering resources, provides guidance on systems engineering processes, and participates in systems engineering activities throughout The MITRE Corporation.

Systems engineering resources are available through the SEPO Library, which contains a broad spectrum of information and knowledge to help you on such topics as acquisition, systems engineering, software engineering, decision support, and process management. If you are looking for a specific item, you can search the SEPO Library using MITRE Google or subscribe to subject updates to find out when new items are added to the SEPO Library in a specific subject area

We offer systems engineering guidance through our SEPO Toolkits. Current toolkits available online include *Risk Management*, *Requirements Process*, *Configuration Management*, *Program Assessment*, *Enterprise-Node-Program Management*, and *Multinational Interoperability*. If you want guidance in another area, such as Software Engineering, Acquisition, the Capability Maturity Model–Integrated (CMMI) Process, or sponsor-specific systems engineering areas, please contact our team.

SEPO welcomes feedback on our resources and services. Please feel free to contact us at sepo@mitre.org with your suggestions.

This newsletter is a publication of the Systems Engineering Process Office (SEPO). Additional copies are available.
The MITRE Corporation, 202 Burlington Road, Bedford, MA 01730-1420 and 7515 Colshire Drive, McLean, VA 22102-7508