The US Military Health System encompasses over 60 information subsystems. Keeping it on track requires an orderly, well-crafted enterprise architecture.

Raj Mukherji, Csaba Egyhazy, and Marco Johnson

Architecture for a Large Healthcare Information System

An enterprise architecture is a collection of models, diagrams, tables, and narrative that together translate a given entity's complexities into a simplified, yet meaningful representation of its operation. In simplest terms, an enterprise architecture offers a way to describe the structural composition of business activities and automation systems.

In the US, according to the Clinger-Cohen Act of 1996 (http://www.4.law.cornell.edu/uscode), when a federal agency acquires a large automation system, the CIO must develop and maintain an enterprise architecture and use it to make investment decisions. The Clinger-Cohen Act also mandates that all federal information systems align with the organization's strategic goals and objectives. Ensuring that alignment requires a well-developed enterprise architecture.

We developed one of the three major components of such an enterprise architecture, the operational architecture, for the US Department of Defense's Military Health Systems (MHS). The MHS is an enormous operation: Its electronic patient record (the Composite Healthcare System) in conjunction with other supporting systems, creates, updates, and accesses medical records for many millions of Department of Defense personnel and their families worldwide. As of 2002, the US Army, Navy, and Air Force manage as many as 154 medical treatment facilities worldwide. This article reports on our real experience and the lessons learned as we developed this system's operational architecture.

Enterprise Architecture Maturity Model: The Five Levels

1. **Creating enterprise architecture awareness**
2. **Building the enterprise architecture management foundation**
3. **Developing architecture products**
4. **Completing the architectural products**
5. **Leveraging the architecture for managing change**

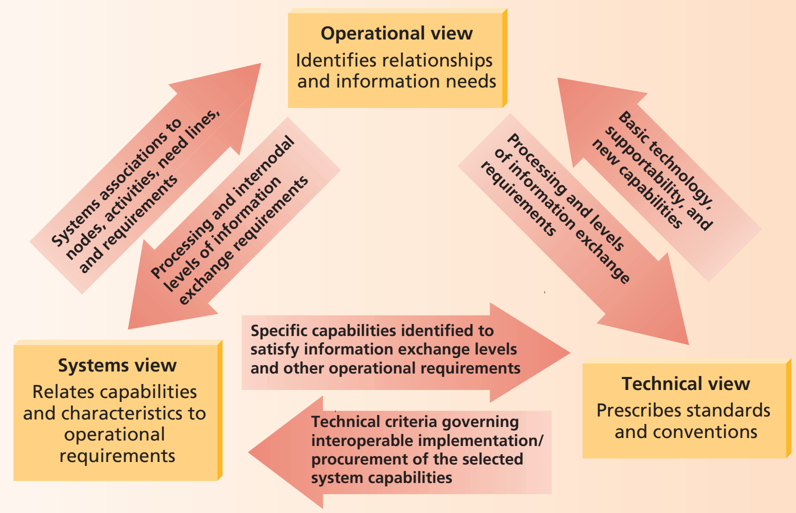
BACKGROUND: ENVIRONMENT AND ARCHITECTURE

Any enterprise architecture can be assessed against the five-level Enterprise Architecture Maturity Model developed by the US Government Accounting Office (*Architecture Across Federal Government,* GAO-02-6, Government Accounting Office, Washington, D.C., 2002). According to this model, summarized in the "Enterprise Architecture Maturity Model: The Five Levels" sidebar, an enterprise architecture reaches its highest stage of maturity—level five—when it becomes the organization's tool for managing change across its systems. Our experience was with an architecture at the third stage of maturity—in the process of developing architecture products.

Architecture development begins with a given set of forward-looking user requirements, legacy

1520-9202/02/$17.00 © 2002 IEEE

Figure 1. Relationships among the operational, technical, and systems views are key to developing and deploying interoperable information systems.



hardware systems, applications, databases, and operating systems. In the MHS environment—which uses a geographically distributed, worldwide-deployed information system—developing, documenting, and maintaining the enterprise architecture takes place in six-month cycles, with a new version of the architecture published in each cycle. To quickly assess the impact of changing requirements on this complex architecture as a whole, we had to rapidly assess—taking just a few weeks—the impact of new or changed requirements on the existing large, complex legacy systems.

In theory, CIOs should use architecture to project dollar impacts on existing systems; from there they can derive project costs, schedules, and budgets. However, generating systematic, rigorous impact analyses of changing requirements on complex systems is difficult. This is particularly true when the available architecture is at stages one to three in the five-stage framework for architecture maturity.

In addition to providing a means of analyzing the costs of changing requirements on complex systems, an enterprise architecture can serve several other beneficial roles:

- facilitating the use of a common language for architectural products,
- supporting reuse of standardized components,
- enhancing the potential for standards reuse, and
- serving as a training tool for management and contractors.

The MHS' mission is to deliver healthcare to Department of Defense personnel and their dependents worldwide. Composed of 60 large information subsystems, MHS supports business processes in four core areas:

- making healthcare accessible,
- providing healthcare,
- managing population health, and
- managing the business itself.

Simply developing the systems requirements for the MHS took an 18-month effort by a multidisciplinary team that included healthcare specialists—doctors, nurses, surgeons, technicians, pharmacists, dentists, and administrators—as well as logisticians, mobile deployments of healthcare specialists in theatre locations, and information scientists. We used the requirements this team developed to derive the operational architecture.

To start defining an operational architecture, we started with the architectural framework for Command, Control, Communications and Computers, Intelligence, Surveillance and Reconnaissance—known as C4ISR (see http://www.c3i.osd.mil/org/cio/i3/AWG_Digital_Library/index.htm). The C4ISR framework is generic enough that it lends itself to a wide variety of application domains outside the Department of Defense. For example, the US Federal Aviation Administration uses similar architectural artifacts to test and develop avionics.

C4ISR guidance requires documenting both the current (as is) and future (to be) configuration of computing resources, business rules, and relationships among operational entities and the supporting information systems, with explicit references to standards and technologies. C4ISR has 26 different associated architectural artifacts, some optional, others mandatory.

Our operational architecture drew on these C4ISR artifacts to come up with operational-architecture artifacts to describe the MHS business. From the generic C4ISR architectural framework, we described operational views for the MHS using our own symbolic language for node connectivity diagrams and for matrices defining information exchange requirements.

VIEWS AND THEIR INTERCONNECTIONS

C4ISR provides guidance for describing information and organizational systems and their activities in the context of the enterprise mission. To achieve this architectural description, the framework combines three perspectives: the *operational, systems,* and *technical* views. Figure 1 shows the interrelationships among these three views.

In addition to the operational, systems, and technical views, the C4ISR framework includes a fourth perspective—called

Table 1. Mandatory and optional operational-architecture products.

Product reference	Architecture product	Mandatory or supporting	General description
OV-1	High-level operational concept description	Mandatory	High-level graphical and textual description of operational concept (high-level organizations, missions, geographic configuration, connectivity, and so on)
OV-2	Operational node connectivity description	Mandatory	Operational nodes, activities performed at each node, connectivities, and information flow between nodes
OV-3	Operational information exchange matrix	Mandatory	Information exchanged between nodes and the relevant attributes of that exchange, such as media, quality, quantity, and required level of interoperability
OV-4	Organizational relationships chart	Supporting	Command, control, coordination, and other relationships among organizations
OV-5	Activity model	Mandatory	Activities and relationships among activities, inputs, and outputs. In addition, overlays can show cost, performing nodes, or other pertinent information
OV-6a	Operational rules model	Supporting	First of three products describing operational activity sequence and timing. This product identifies the business rules that constrain the operation.
OV-6b	Operational state transition description	Supporting	Second of three products, this product identifies a business process' responses to events.
OV-6c	Operational event/ trace description	Supporting	Third of three products, this product traces the actions in a scenario or critical sequence of events.
OV-7	Logical data model	Supporting	Documentation of the data requirements and structural business process rules of the operational view

all views—that summarizes the enterprise's mission and goals.

Operational view

The operational view contains graphical and textual descriptions of the operational nodes and elements, assigned tasks and activities, and information flows required between nodes. It defines the types of information exchanged, the frequency of exchange, which tasks and activities these information exchanges support, and a characterization of the information exchanges with enough detail to indicate specific interoperability requirements.

Systems view

The systems view shows how multiple information systems link and interoperate; it can also describe the internal construction and operations of particular information systems within the architecture. In addition, it can identify key hardware and software, and define the hardware's physical connections and locations. This view can also include data stores, circuits, and networks, and it can specify system and component performance parameters. The systems view associates physical resources to the operational view and its requirements, according to standards defined in the technical view.

Technical view

This view provides implementation guidelines for the technical systems. Project participants use these guidelines to prepare base engineering specifications, establish common building blocks, and develop product lines. The technical view includes a collection of the technical standards, conventions, rules, and criteria. The view further organizes this collection into profiles that govern system services, interfaces, and relationships for particular systems views. These profiles also relate to particular operational views.

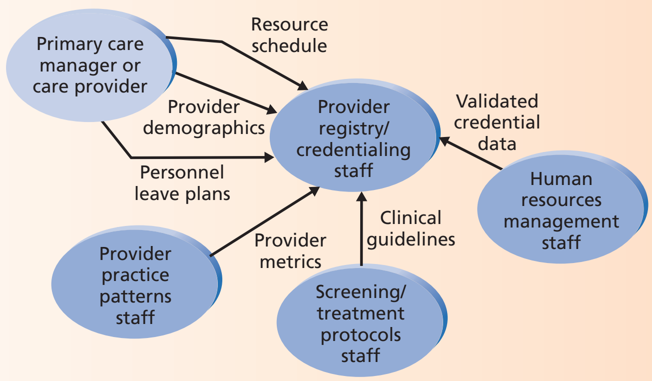
All views

The all views perspective consists of two architectural products—AV-1 and AV-2. In the initial phases of architecture development, AV-1 serves as a planning guide. Upon an architecture project's completion, it provides a textual summary of the project's who, what, when, why, and how. AV-2 provides a central source for all definitions and metadata, including those that other products might also provide for convenience.

CREATING VIEWS: WHERE TO START

Each architectural view in Figure 1 consists of sets of subviews, which C4ISR further categorizes as either mandatory or supporting types. Table 1 shows the operational view's subviews and their categorizations, which our architecture inherited from the C4ISR framework. Although C4ISR provided the categorizations, it provided no guid-

Figure 2. OV-2 example: Node connections and information exchange requirements for establishing and maintaining schedules.



ance about the order in which to implement the products. Thus, our first question was "Where do we begin?"

The key to rapidly developing operational architectural depictions is to selectively choose the order of the mandatory architectural-framework products. Our experience showed that implementing the mandatory operational views in the following order made it relatively easy to derive the operational architecture from functional requirements:

1. OV-5, activity model;
2. OV-1, high-level operational concept description;
3. OV-2, node connectivity diagrams; and
4. OV-3, information exchange requirements.

We advise doing OV-5, the activity model, first because it is the starting point for deriving OV-1, the OV-2s, and the OV-3s. OV-5s are typically textual descriptions of business processes. OV-1 is independent of the other mandatory artifacts in that it merely summarizes the activities in a high-level diagram. However, we derive OV-1 from a summary of OV-5 at the enterprise mission level. We can generate OV-2s (also known as node-connectivity diagrams) only after the business processes are well understood and documented. OV-3s, which we derive from OV-5s and OV-2s, require detailed understanding of the information exchanges between business entities in the OV-2 diagrams. An OV-3 captures the details of the exchanged information as a matrix of information exchange requirements.

SYMBOLIC REPRESENTATIONS FOR OPERATIONAL ARCHITECTURE

As we mentioned, OV-5s are typically textual descriptions; OV-1s might use PowerPoint or some other visual tool to create representations of the enterprise's missions and interfaces to external systems. But there are many different methods available for creating the node connectivity diagrams and information exchange requirements (OV-2s and OV-3s). Architects must choose a representation method suited to their environment.

The C4ISR guidance gives samples of each operational view. But for the MHS environment, we had to extend the guidance for generating OV-2s and OV-3s so we could represent the external agencies that interact with MHS entities.

OV-2: Node connectivity diagrams

In a node connectivity diagram, a node is an operational-architecture element that produces, consumes, or processes data or information. In addition, according to version 2.0 of the C4ISR architecture framework, operational nodes perform a role or mission. Operational-architecture node connectivity diagrams are an extension of traditional data flow diagrams (DFDs), but their nodes differ from those in traditional DFDs in two subtle ways:

- Besides acting as data sources and sinks, nodes in OV-2

Sample Beneficiary Scenario

During Lieutenant John Smith's in-processing, a Navy office clerk begins to collect a core set of information. John states that he is concerned about his family's healthcare benefits because his son, Mike, has a preexisting condition—pediatric asthma. The Navy office clerk immediately shares with John three different health plans that his family is eligible for. The Navy office clerk points out that John and his family must be assigned a primary care manager (PCM) if they choose Tricare Prime. He prompts the composite health care system to respond with photographs of and information about several PCMs who fit John's prerequisites and are located at a medical treatment facility close to John's home. Lt. Smith explains that he would like to take the information home and have his wife, Maria, help him choose. Before John leaves, the Navy office clerk asks John the best way to contact him or Maria. John says either by phone call or his personal e-mail.

John goes home and discusses enrollment and PCM options with Maria. During the discussion, he accesses a program that explains the three different health plans. To assess the choices, he uses the Tricare Personal Healthcare Consultant. John completes the calculator inputs by selecting the most appropriate answers to the questions (location, age of beneficiaries, services used in the past, and so on), and the Personal Healthcare Consultant returns the best choice for a health plan based upon his requirements. Several days later at home, John receives an e-mail message that reminds him that the processing is not complete, as he has not communicated his decision to enroll or select a PCM. John replies that he wants to enroll into Tricare Prime but still has a few more questions.

The Navy office clerk reads John's e-mail response and calls him to schedule an appointment to complete in-processing and enrollment. Several days later, as scheduled, John and his family go to the Navy personnel office. The Navy clerk brings the Smiths into his office. After asking a few more questions, John and Maria accept Commander Jones as their PCM. During the meeting, the system notifies the Navy office clerk that Mike will also have a pediatric case manager because of his pediatric-asthma diagnosis. The Navy office clerk prints out a customized healthcare benefits package, which lists benefits for John and his family, the assignment and location of the family's PCM (Cdr. Jones), the assignment of the pediatric case manager (Lt. Cdr. Kim Snow), what numbers to call with questions, and where to go in an emergency (the Triage Center).

The Navy office clerk also directs John to the Web sites maintained by the medical treatment facility and issues the whole family national medical ID cards. The Navy office clerk also tells the Smiths that the MHS offers a healthcare orientation class and explains how they can register.

diagrams can also process data.

- Nodes in OV-2 diagrams can also process data besides producing or consuming data. In C4ISR terminology, nodes can be *actors* in the object-oriented modeling sense.

In a traditional DFD, the data transferred between nodes is usually labeled but has no detailed attributes associated with it. Node connectivity diagrams label the data exchanged between nodes but also explicitly record many key attributes associated with the data. This record is in tabular form (OV-3s).

In the MHS to-be operational architecture, internal nodes are organization types or roles that perform business processes or activities within the MHS scope. They do not represent automated systems, existing organizational, or force structures. External nodes are organizations outside the MHS scope that provide or obtain MHS-related information in support of MHS business processes. We had to extend the C4ISR's OV-2 definitions to include separate representations for external nodes.

The C4ISR guidance for creating OV-2s yields operational node connectivity diagrams such as that in Figure 2, which we generated using the following procedure:

1. Use OV-5, the activity text descriptions, and associated scenarios to identify the terminal nodes. (For an example, see the "Sample Beneficiary Scenario" sidebar.)
2. Identify what information the system produces and consumes.
3. Connect nodes using information exchanges described in scenarios and activity descriptions (OV-5s).
4. Describe the information exchange requirements in the OV-3s.

The scenarios are key to establishing system requirements and standards. They help define the roles of various participants in the system. Scenarios also provide a sim-

How a Scenario Aids System Design: Attributes, Standards, Roles, and Rules

Scenarios provide a useful way to simply convey certain system attributes. Here, we describe how our sample scenario defines important system qualities.

Beneficiary Scenario Attributes

This particular scenario defines several system requirements, including

- system prompts to ensure that the enrollment specialist collects the appropriate information,
- access for enrollment specialists and beneficiaries to a central source of PCM and physician profiles,
- identification of an appropriate PCM based on user preferences,
- prompts to verify specific registration and insurance information fields,
- access to enrollment and eligibility information (central access to patient data and information),
- access to education tools and materials,
- ability to issue a standard medical card,
- access to health education medical treatment facility Web sites,
- ability to generate and print customized (tailored) marketing/education package based on health plan,
- identification of family-specific enrollment options,
- assignment of a PCM or case manager during enrollment,
- education for enrollees about their benefits at the time of enrollment,
- a provider directory containing internal and external provider information, and
- the ability to create a permanent provider file.

Beneficiary Scenario Standards

System view architects use the following standards to define more detailed system requirements:

- standard set of enrollment information,
- interoperable systems that support messaging of care team members and administrative personnel,
- common set of patient education tools,
- medical treatment facility Web sites displaying up-to-date information,
- enrollment integrated into in-processing,
- single patient identifier,
- single provider identifier,
- a comprehensive clinical and financial data repository,
- decision support tool(s) that assist individuals in selecting the most appropriate health plan (using, for example, branching logic), and
- standard responses to beneficiary inquiries.

Role Requirements

Covering enrollment and eligibility requires personnel for several positions, each of which has specific requirements:

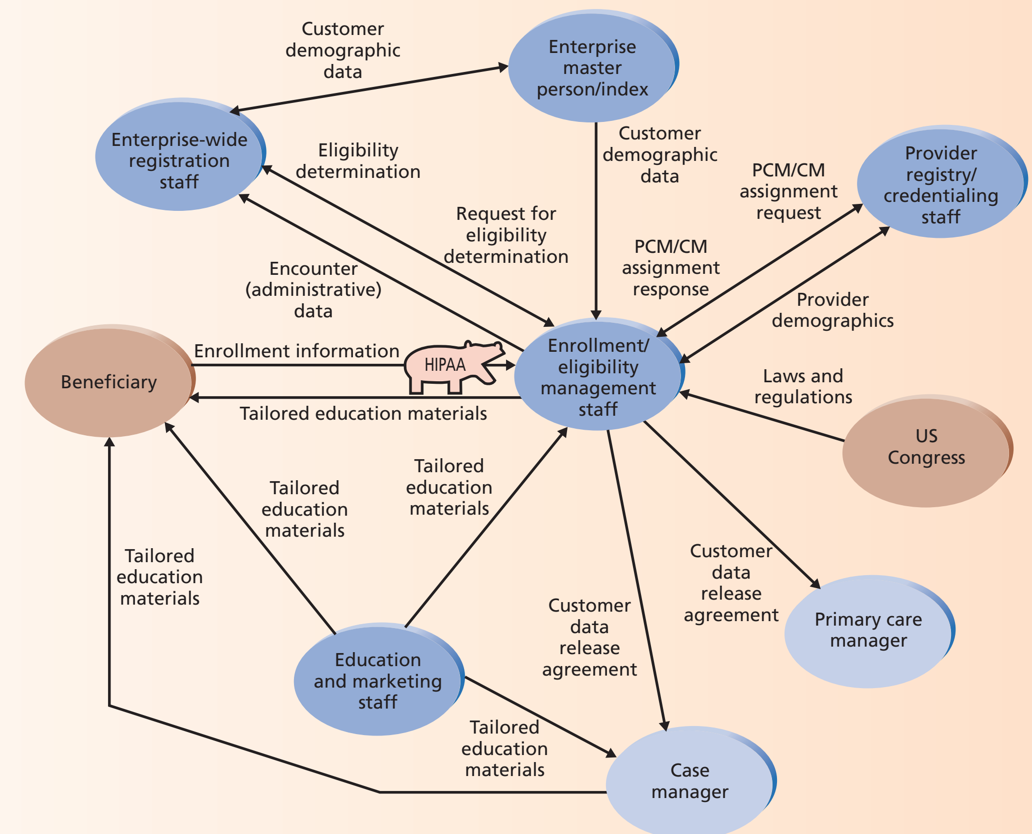
- The *Webmaster* has a competency-based customer service orientation and uses staff and tools to maintain the Web version of the composite healthcare system.
- The *enrollment specialist* is selected from service-oriented personnel (courteous, competent, and caring individuals) and uses knowledge of composite healthcare benefits and eligibility requirements to enroll Department of Defense personnel.
- The *primary care physician* is educated in managed care and MHS and uses the composite health care system to access, provide, and record medical information related to care provision.

Beneficiary Scenario Business Rules, Policies, and Practices

The MHS medical and business personnel, including treatment facility managers, establish the following rules, which the system must adhere to:

- During the enrollment process, enrollment specialist staff assigns all patients or families a PCM.
- Enrollment occurs in real time.
- A beneficiary's PCM must be an in-network or contracted provider.
- In-processing is not complete until enrollment is complete.
- The enrollment staff assigns case managers as necessary, based on established criteria (for example, high-risk or high-cost diagnoses).
- Providers must update their profiles regularly.
- Medical-record information transfer incurs no down time.

Figure 3. OV-2 example: Node connections and information exchange requirements for establishing, maintaining, and accessing enrollment and eligibility. Blue, brown, and light-blue nodes represent extensions to the C4ISR framework.



ple way to convey business rules, policies, and practices. The "How a Scenario Aids System Design: Attributes, Standards, Roles, and Rules" sidebar discusses how our example scenario conveys these key attributes.

Extending OV-2

The steps we just enumerated produce diagrams consistent with the C4ISR guidance. However, we found that the guidance did not satisfy some of our architecture's requirements; it did not provide

- nodes that differentiate activities and systems internal to the DoD from those outside it;
- nodes that are unique to MHS, such as primary care physicians;
- large data stores with unique legal and DoD requirements for storing large amounts of patient-related information; and
- node connections that can include legislatively driven information, such as the information required by the Health Insurance Portability and Accountability Act (HIPAA).

To handle these unique requirements, we extended C4ISR's OV-2 guidance by including colors to differentiate between four node types:

- internal (blue),
- external (brown),
- primary care physician (light blue), and
- data repositories (light green).

Figure 3—an operational node connectivity diagram for establishing, accessing, and maintaining enrollment eligibility of DoD staff—illustrates this extension. This particular diagram does not include a light-green node, but readers can see this node type in many of the operational architecture's other 97 node connectivity diagrams (http://www.hmrha.hirs.osd.mil/Prod/0root.htm).

In Figure 3, arrows indicate information transfer

Table 2. OV-3 example: Information exchange requirements for establishing, maintaining, and accessing enrollment and eligibility.

Information name	Description	Media type	EIA data entity	Source node	Destination node	Security classification	Timeliness	Criticality	Frequency
Customer data release agreement	Authorization permitting the provider to access and use identity-linked healthcare information	Data, text	NA	Enrollment/ eligibility management staff	Case manager	SBU	Seconds, hours, or days	High	Event driven
				Enrollment/ eligibility management staff	Primary care manager	SBU	Seconds, hours, or days	High	Event driven
Customer demographic data	Facts about the beneficiary population, such as address, phone number, occupation, and so on	Data	Patient demographic	Enterprise-wide registration staff	Enterprise master person index	SBU	Seconds	Activity critical	Event driven
				Enterprise master person index	Enterprise-wide registration staff	SBU	Seconds	Activity critical	Event driven
				Enterprise master person index	Enrollment/ eligibility management staff	SBU	Seconds	Activity critical	Event driven
Eligibility determination	Assessment resulting from evaluation of an individual's eligibility criteria	Data, text	NA	Enrollment/ eligibility management staff	Enterprise-wide registration staff	NA	NA	NA	NA

between nodes, with labels indicating the type of information transferred. Details of the format, type, and other characteristics of each information transfer are listed in OV-3 tables—information exchange requirements (IERs)—covered in the next section. Double-ended arrows indicate exchange in both directions for a pair of interconnected nodes. Many-to-many interactions are permissible; interactions can be sequential or simultaneous.

Arrows accompanied by a hippopotamus icon indicate information exchanges that require the HIPAA data transfer format (http://www.hirs.osd.mil/HA). A cloud over an arrow (not shown in Figure 3) indicates Internet use. The default interaction between nodes is client-server.

OV-3: Information exchange requirements

OV-3s consist of specifications for IERs, which describe requirements for the information flow between nodes. This description is a table that identifies nodes and their associated information or data requirements. An IER typically includes attributes such as information size, throughput, timeliness, quality, and quantity.

As the next level of architectural detail down from OV-2s, IERs include data attributes for each node in an OV-2 diagram. Table 2 is a sample IER for Figure 3, the node connectivity diagram for establishing, maintaining, and accessing enrollment and eligibility. The EIA data entity column contains the Enterprise Information Architecture Data entity. The source node sends the information, and the destination node receives it. In the OV-3 context, the mechanisms for delivering the information are unimportant. The OV-3 explicitly identifies only the roles as seen in the OV-2s and their information exchange requirements. The exception to this rule for OV-2 diagrams is for HIPAA and Internet use. In the security classification column, SBU stands for sensitive, but unclassified.

In addition to the name and description, Table 2 lists eight attributes for each information exchange. These attributes are key to determining system performance and design issues. The operational-view architects hand off both the OV-2s and OV-3s to the system view architects for further detail generation using the C4ISR-mandated system view products.

LESSONS LEARNED

Our experience using the C4ISR architectural framework to develop operational views for the Military Health Service led us to conclusions on three main topics.

Ease of use

System architects have implemented the C4ISR architectural framework using a wide variety of symbolic representations, as evidenced in the Composite Health Care System II, Army Medical Directorate, and Joint Electronic Commerce Program Office architectural models. CASE (computer-aided software engineering) tools have also worked well for implementing these symbolic representations of architectural models—for example, Casewise Modeler 2000, Rational Rose, Visio, and PowerPoint.

Our experience convinced us that the primary driver for choosing a symbolic modeling language and tool is its simplicity for modeling business processes. This is because operational views are primarily concerned with business models and workflows for information exchanges among functional entities.

Limitations of the approach

We have emphasized streamlining the selection and development of mandatory C4ISR operational views to minimize risk exposures from possible schedule, cost, and quality failures. The products developed using this approach must give system architects, designers, and application developers an easy transition to their different architectural and design products. Traceability from the operational views to the actual implementation is certainly a desirable quality for any architecture.

When we started this project in October 1999, no off-the-shelf tool satisfied our need for simplicity of use by multidisciplinary teams. A major reason we ended up choosing the Casewise 2000 Modeler tool for capturing the activity models and mandated operational-view products was the perception that we should reuse legacy documents as much as possible. Modeler 2000 allowed more reuse than other candidate tools, such as Rational Rose.

Future work

We could extend our work by providing mappings to other common architectural frameworks such as RM-ODP—the reference model for open distributed computing—as described by Csaba Egyhazy, Scott Eyestone, and Janet Martino, ("Object-Oriented Analysis and Design: A Methodology for Modeling the Computer-Based Patient Record," *Topics in Healthcare Information Management,* Aug. 1998, pp. 48-65); the NIST/ECMA model (from the US National Institute of Standards and Technology and the European Computer Manufacturers' Association); and the three-layer enterprise architecture planning model described by Steven Spewak and Steven Hill (*Enterprise Architecture Planning,* Wiley Interscience, Boston, 1992). ■

Raj Mukherji *is a senior consultant at the US Department of Defense's TriCare Management Activities, Electronic Business, Policy and Standards Directorate, Office of the Assistant Secretary of Defense/Health Affairs. Contact him at Raj.Mukherji@tma.osd.mil.*

Csaba Egyhazy *is an associate professor in computer science at Virginia Tech. He teaches at the Northern Virginia Graduate Center, Falls Church, Virginia.*

Marco Johnson *is Director of eBusiness Architecture, Electronic Business, Policy, and Standards Directorate, Military Health Systems, Office of the Assistant Secretary of Defense/Health Affairs.*

We acknowledge the assistance of Captain Brian Kelly, US Navy; Sherry McKenzie; Nancy Orvis; David Orris; Steve Lichtman; and Al Obuchowski.

Get access

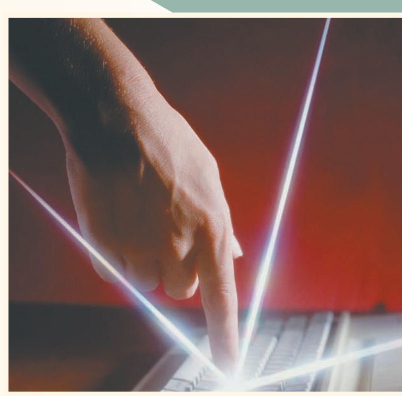
to individual IEEE Computer Society documents online.

More than 57,000 articles and conference papers available!

US$9 per article for members

US$19 for nonmembers

http://computer.org/publications/dlib/



IEEE Computer Society