Carderock Division
Naval Surface Warfare Center
West Bethesda, Maryland 20817-5700

CDNSWC-TSSD-98-10 August 1998
Total Ship Systems Directorate
Research and Development Report

PROPOSED WEB-BASED JOINT IETM ARCHITECTURE (JIA) FOR THE INTEROPERABILITY OF DoD IETMs.

by

Eric L. Jorgensen

Proposed Web-Based Joint IETM Architecture (JIA)

for the

Interoperability of DoD IETMs

by

Eric L. Jorgensen

August 1998

Naval Surface Warfare Center
Carderock Division, Code 2052
West Bethesda, MD 20817-5700

ABSTRACT

This Report documents the results of a study sponsored by the Assistant Deputy Secretary of Defense for Logistics Reinvention and Modernization (ADUSD (LR&M)) and the Joint Commanders Group for Communication and Electronics (JCG-CE). An Architecture is proposed for Interactive Electronic Technical Manuals (IETMs) based on the technology, industry standards, and commercial software products being developed for the World Wide Web

The objective of the DoD Study Effort is to create an IETM Architectural Framework, which fosters acquisition management policies, and procedures that guide and standardize IETM acquisition, management, and display. The scope of this study and the resultant Architecture is end-user interoperability that:

- will enable maximum interoperability of Technical Information displayed and accessed by an end-user in meeting the needs of the Defense Logistics community in supporting the material readiness of the DoD forces;

- serve as the basis for a formal DoD-wide adoption of the proposed approach in promulgating the required acquisition and field-support policy, to be based on a series of FY99 pilot-demonstration programs that will show the applicability and efficacy of the Architecture in supporting IETMs for a broad spectrum of candidate weapon systems of the Military Services.

The Report documents the recommended Web-based Architecture, called the Joint IETM Architecture (JIA), including a summary of what will eventually be four Performance Specifications for the following areas:

- Object Encapsulation and Component Interface.

- Intranet Server and Database Interface.

- Common Browser.

- Electronic Addressing.and Library Model

ACKNOWLEDGMENTS

The author was assisted in the preparation of this document by the technical contribution and review of members of the Tri-Service IETM Interoperability Project Team, which he leads. The actual technical input upon which much of this Architecture is based will be documented in a series of separate reports. Principal technical-team contributors for the JIA Effort include the following:

NSWC, Carderock Division - John Junod

ManTech Advanced Systems International, Inc., West Virginia - Bob Kidwell, Joe Brazy, Don Reynolds, Glen Copen, Tom Morris

ANTECH Systems - Dave Cooper

AERA - Glenn Handrahan

PBM Associates - Pushpa Merchant

BTAS - Gary Forrester

Funding for the NSWC portions of this Effort came from the Office of the Assistant Deputy Undersecretary of Defense (Logistics Reinvention and Modernization).

Performing Activity: NSWCCD Code 2052 Bethesda MD - Eric L. Jorgensen (jorgense@dt.navy.mil)

Table of Contents

This Page Intentionally Left Blank

1. INTRODUCTION

The transmission of digital data within the Department of Defense (DoD) is quickly becoming the dominant method for communicating and accessing the Technical Information needed to operate and maintain military weapon systems required to support field operations. In response to directives from the Office of the Secretary of Defense, all of the Military Services have ongoing efforts to convert paper-based technical documentation into digital format. They are replacing existing maintenance and logistic-support Technical Manuals with legacy-data-conversion products in the form of Electronic Technical manuals (ETMs) and the newer Interactive Electronic Technical Manuals (IETMs). Since this information is needed to sustain war-fighting capability in Joint and multi-unit operations, a uniform approach throughout DoD must be developed for managing, fielding, and viewing the digital products. This Report is the result of the initial phase of a DoD-wide study conducted in response to a requirement of the Joint Logistics Commanders (JCG-CE) to develop a common user interface for this digitized information.

This Report recommends that a new coordinated procedure of acquiring and deploying IETMs replace the current practice of independent procurement of Technical Information using divergent technologies and stand-alone formats. This new process would be guided by an overarching technical Architecture that permits the IETM applications to interoperate and work together at the user interface, without requiring that all programs employ the same IETM implementation, authoring system, or support infrastructure. A general IETM Architecture is described for development and deployment of IETMs throughout DoD. As each individual IETM selection must, of course, be evaluated according to individual Program and Service parameters and restraints, the Architecture permits a wide range of solutions and specific implementations so that procurements can be based on sound business decisions. The objective of the Architecture is that, regardless of the source and peculiar format of that source data, the benefits of interactive Technical Information can be provided to any war-fighter for viewing and utilizing the Weapon System support data. This is accomplished by utilizing a common user interface to any JIA-compliant IETM, as well as, by automating the selection of that information from a uniformly assessable electronic technical-library resource. Such a common process for managing and deploying digital data will make the most effective use of existing resources and will provide vitally needed IETM interoperability.

This Report represents a comprehensive but preliminary release of the JIA. The final release resulting from the current DoD Effort will not be available until the task is completed in June 1999.

1.1 The Problem

In 1992 the DoD issued three Military Specifications for Service-wide use in the acquisition of IETMs. These specifications have been successful in their original objective of guiding the development of IETMs, which are now being acquired for many of the DoD's new major weapon systems. However, as individual systems have matured, issues in the area of

interoperability between the differing IETM presentation systems have arisen. Additionally the Services have noticed substantial incompatibility between these IETM systems and the growing inventory of legacy-data Electronic Technical Manual (ETM) systems (to which the Specifications were not directed). With the new IETM format, nearly all early developers of Specification-compliant systems had to create both a new authoring system and a new user-presentation system, since they had no existing software products on which to build their development. The net result was that the authoring system and the presentation system developed for an individual IETM were interdependent, but were designed independently of other IETM or legacy-based ETM systems. Thus, an IETM authored by one activity usually could not be viewed using a presentation system developed by another activity, nor could it electronically reference or include the legacy-ETM information in the same presentation system.

Initially, this situation was not a problem for a weapon-system Acquisition Manager who acquired IETMs, because the developer, typically a prime contractor, was able to control both the IETM and the display system for the dedicated user population for any particular weapon system. But, as the use of IETMs became more widespread and they began to be deployed at multiple sites, it became important to establish a consistent infrastructure to manage and distribute IETM updates to the field sites and to provide life-cycle support for numerous types of IETMs. In this environment, the fact that differing IETMs cannot interoperate (i.e., cannot be viewed on the same standard presentation system, or electronically reference each other to any meaningful level of internal granularity) has become a major impediment.

An additional problem area has resulted from the lack of a common standard for the structure of the delivered IETM in its digital packaging: it has been very difficult if not impossible to define the requirements for, or to make the initial design of, a standard information infrastructure to support IETMs in the field.

1.2 Expanding an Existing Study for Naval Aviation to all of DoD

In late 1986, the Naval Air Systems Command recognized these problems and, acting in accordance with the emerging Navy Logistics Information Strategy Plan (formally published 8 July 1997), initiated a major study to develop a Navy IETM Architecture (NIA) that, when implemented, would facilitate electronic Technical Information interoperability among Naval Aviation Programs. This Effort was funded by the Naval Air Technical Service Facility and was conducted under the technical leadership of the Naval Surface Warfare Center, Carderock Division (NSWCCD), Code 2052, Bethesda MD. This first study was completed in April 1998 and the NIA document was presented to NAVAIR Policy Officials who will decide the extent to which the recommendations contained in this Report will become NAVAIR Policy or required practice.

Subsequent to the Navy Effort, the DoD Tri-Service IETM Technology Working Group (IETMTWG), chartered by the OSD CALS Office of DUSD(L), endorsed the NAVAIR Project as an approach offering the potential for a DoD-wide solution to the interoperability

problem. At the request of the OSD CALS Office, the IETMTWG expanded the NAVAIR project plan and approach into a DoD-wide Effort that involves modifying, prototyping, and testing the NAVAIR initiated interoperability methodology. The approach was to utilize an expanded set of Tri-Service requirements and demonstrate the DoD Architecture on a spectrum of DoD weapon systems. At the same time, the proposed IETMTWG plan was presented to the Technical Publications Sub-panel of the Joint Commanders Group for Communications and Electronics (JCG-CE) as a means of meeting some of the major goals of the JCG-CE Publications Panel. These goals included the achievement of field interoperability for IETMs. The proposed approach was approved and the JLC recommended, by a memorandum of 10 June 1997, that the OSD CALS Office implement this plan as a joint Effort between the JCG-CE and the IETMTWG. This DoD-wide effort technically started in late 1997 and will continue through June 1999. NSWCCD is also leading this DoD Effort. The OSD CALS office has been reorganized as ADUSD(LRM) and remains the chartering activity for the IETMTWG and this project.

1.3 Objective of the Study

The objective of the DoD Study Effort has been to create a high-level Joint IETM Architecture (JIA) to guide and standardize IETM acquisition, management, and display that:

(1) will enable, for the end user, maximum interoperability in the use of Technical Information to meet the needs of the Defense Logistics community in supporting the material readiness of the DoD forces; and

(2) will also serve as the basis for a formal DoD-wide adoption of the proposed approach in promulgating the required acquisition and field-support policy. To reduce the risk of implementation and to demonstrate utility of the approach, the policy recommendations will be based on a series of FY99 pilot-demonstration programs that will show the applicability of the Architecture to support IETMs for the whole spectrum of weapon systems of the Military Services.

1.4 Goal for the Architecture

The primary goal for the JIA is to establish a technical framework for acquisition and deployment of the whole spectrum of Electronic Technical Manuals, so that when the sharable and interoperable Technical Information is distributed to the work location of an end-user, he will be able to view and utilize that data through a common user interface, no matter what the authoring source or data format. In so doing, the DoD will be able to establish a unified approach to the acquisition, management, and use of existing ETMs and newly procured IETMs. To meet this goal, the overall approach will be based on the use of existing COTS (Commercial Off The Shelf) and NDI (Non-Developmental Item) Internet and World Wide Web technology. An overall end goal is to achieve end-user-level interoperability of the IETMs delivered to and used by the entire DoD Operational Community. In this context, an ETM or IETM is defined as having end-user interoperability

when it can enable a user with one common, commercially available display device, such as a portable personal computer:

(1) to view and interact with Technical Information from any source and of any internal format; and

(2) to automatically access and view, by means of an electronic-link reference in the displayed Technical Information, additional information in any other ETM or IETM with which the link connects him.

1.5 Purpose and Scope of this Report

The JIA has been developed to provide interoperability for all levels of Electronic Technical Manuals including all five established DoD ETM/IETM Classes from the digitized page-oriented TMs to the highly integrated Interactive Electronic Technical Manuals. For purposes of this Report and the recommendations contained herein, the term "IETM" will hereafter be used to refer to all classes of ETM/IETMs whether the existing class definitions call them ETMs or IETMs.

The purpose of this Report is to describe the portions of the joint IETM Architecture applicable to end-user interoperability so that three major constituencies can develop needed capabilities for a Interoperable IETM End-User Capability. The three targeted constituencies are:

(1) The creators of the IETM products (both content providers and presentation-software vendors);

(2) The developers of the IETM user-infrastructure for both the distribution infrastructure and the user site intranet; and

(3) The procurers of the common display devices with the common JIA-compliant browser software installed on the devices.

The Report is also intended for the use of DoD Logistics Managers and Acquisition Program Managers who are responsible for policy and direction of these constituencies.

1.6 Technical Approach

The overall concept of this Effort is to utilize the group of emerging technologies that the commercial marketplace is rapidly adopting as the standard for distributable electronic documents, which are, in general, based on the technology of the Internet and the World Wide Web. For security and operational reasons, the DoD will not, utilize the actual public Internet or the World Wide Web itself, but will employ essentially the same technology and COTS products in a private and dedicated DoD intranet environment. Such an approach is becoming the de facto standard for corporate information-distribution systems worldwide. Once this approach has been proven effective, a set of implementation-guidance documents

and performance specifications will be developed within this comprehensive, DoD-wide, commercially supported (i.e., COTS) framework.

A major objective of the Effort to develop the Architecture is to demonstrate end-user interoperability of proprietary and legacy IETMs. This will be accomplished by encapsulating them into a common View Package (VP) format, which can be electronically distributed to DoD intranets and eventually viewed by an end user employing a single user-information interface (i.e., browser). This process is referred to in this Report as "object encapsulation". Such a demonstration will require the establishment of the following technical capabilities:

(1) an authoring framework to effectively create and manage IETM View Packages for delivery to the Government, regardless of which authoring tools are used;

(2) an infrastructure that permits a Military agency to distribute, manage, and deliver these IETM View Packages; and

(3) a methodology for the end user to access and view the required Technical Information, and to retrieve relevant data from other IETMs, including those of other Services, as necessary.

In order to achieve interoperability, the interface requirements recommended for the JIA will be specific, but they will be constructed so as to encourage innovative and effective solutions, especially in light of the constantly expanding technology base. Achieving this balance has required some decisions that may need to be reexamined over time. Whenever possible, the design will adhere to open standards and/or de facto Internet standards widely implemented by multiple vendors, with the clear intent to maximize the use of commercially available software products.

2. OVERVIEW OF THE ARCHITECTURE

The JIA is firmly based on the proven and widely accepted Internet and World Wide Web technology, implemented as a private Web on a contained intranet. This intranet can be configured as a private DoD World-wide network (e.g., the Global Combat Support System – GCSS), as a combat-capable unit-wide local intranet, or simply as a group of computers in close proximity hard-wired in a local Ethernet configuration. It can also be configured in a single display device (portable or workstation personal computer) which operates as both a client browser and a personal single-user Web server. The technology for implementing such an intranet is low-risk, easily implemented, and widely understood. The proposed Architecture is based entirely on COTS and NDI technology. The Architecture is based on a dedicated Web intranet that has at least one Web-browser client and at least one Web server (more precisely, an HTTP (Hypertext Transfer Protocol) server and its included file-based store), as well as, a network to connect them if they are not contained in the same computer. The specific implementation of the network, which is typically a TCP/IP (Transmission Control Protocol/Internet Protocol)-based network when more than one device is involved, is not discussed in detail in this Report and will typically vary from one implementation to

another. As will be described more fully below, the intranet may include other optional database servers and application servers in addition to the principal HTTP Web-server.

2.1 JTA Compliance

The Joint IETM Architecture will be compliant with the DoD Joint Technical Architecture (JTA). However, the basic Architecture is not intended nor constrained to any specific operating system, and can be adapted to Microsoft NT, DII COE (Defense Information Infrastructure Common Operating Environment), or Unix implementations or a combination thereof for DoD applications. Individual Services or Programs may restrict their IETM applications to one of these operating environments, but neither the JIA nor the JTA require a specific operating system. In technical terms, the "glue" (i.e., the communication protocol) that holds intranet Webs together is the Web protocol HTTP operating over the communications protocol TCP/IP, not the requirement for common operating systems. A TCP/IP network (e.g., an intranet) can easily accommodate multiple operating systems on its server and client computers.

2.2 JIA Use of Internet and World Wide Web Technology

The approach to developing a solution for this interoperability problem has been to adapt commercial and industry applications involving electronic documentation for which there is widespread vendor-product support. A JIA-compliant IETM product will apply the vendor software and standards being developed for the World Wide Web and the Internet in a dedicated and private intranet environment. Following the rapid change trend in Internet technology, the JIA has been designed to be extensible, flexible, and able to accommodate the predictable rapid growth in technology for all aspects of the Internet, the Web, and the emerging electronic documentation applications being developed to operate on the Web.

The Web is, by its nature, a client/server architecture, and there is one area on the client/server spectrum in which JIA-compliant IETM Applications may differ in emphasis from a major server-centric trend that is emerging for many commercial "enterprise" applications. The recommendations for implementation of the JIA are intentionally biased towards a client-centric model employing encapsulated objects that are downloaded to a portable device for use. In this approach, the server is preferred as a utility electronic bookshelf for IETM View Packages (i.e., the package of encapsulated IETM objects). By analogy, these digital books are designed so they can easily be moved to another electronic bookshelf at another physical library site, reflecting the operational reality of the Military unit itself. On the other hand, commercial Web sites tend to be permanently located corporate resource centers at which both the servers and the information providers are located. For these commercial activities, the mobile and less controlled entity is the user client. In these applications, the preference is towards server-centric computing and the use of server-oriented Web-object components. The corporate personnel resources for maintaining both the Web server and the content are located at the Web site. In Military applications, the server sites resemble a technical library rather than a computer information center. The content-related technical expertise lies with the content creator or the end user

and not the administrator of the server. This situation, at this time, dictates total object-encapsulation and client-centric computing as the primary emphasis of the JIA.

Progress in Web-oriented technology and the state of the availability of secure and affordable Military intranets offering Global connectivity may well change this situation in the near future. Thus, the JIA proposed below is intentionally designed so as not to preclude such server-centric solutions should such a change occur. Thus, it is important to emphasize that any implementing policy for the JIA must include some specific guidance on how to apply the Architecture, as well as, the requirement to conform to the Architecture. The use of custom servers is an important issue for which such guidance must be matured over time. Guidance documents for the acquisition of JIA-compliant IETMs must be continually updated over time. Such updates must be based on a continuing study of emerging Military requirements, as compared with the current state of commercial technology and available COTS commercial products. DoD components cannot simply buy the latest commercially available technology without checking it against real Military requirements, which do not always correspond to requirements that govern the creation of commercial products.

2.3 Proposed Requirement Documents for Implementation of the Architecture

This section summarizes initial recommendations for the baseline requirement documents for the JIA and development of JIA compatible IETMs and infrastructure capability.

In addition to requiring the HTTP and TCP/IP networking protocols utilized by the Internet and utilized by virtually all commercial Web-based intranet products and COTS systems, the JIA will be specified in the following areas:

- Object Encapsulation and Component Interface. This specification is needed for definition of the delivery, transport, and structure of the integrated collection of software components and data contained in the IETM View Packages. This specification will include the interface between multiple components, when they exist, and the automated mechanisms for placing the IETM on the targeted intranet. It will also include requirements for the capability to automatically install these components on a user device in a manner sufficiently simple so that no professional system administrator is needed. It will be the primary specification to tell the IETM developers in what form they are to deliver the IETM View Package.

- Intranet Server and Database Interface. For those IETMs that do require the services of an application server and/or a Database server, the IETM supplier must provide the proper software extensions to the basic JIA intranet Web server if they are not already in place. This specification will outline needed cooperation between the constructors of the end-user intranet infrastructure and the IETM provider, and the interfaces and protocols involved. The JIA is designed to recognize the fact that, in certain cases, it will be necessary to install software using conventional system-administration practices on fielded servers in order to achieve needed functionality. (This is not to be the case for the components fielded on JIA-conforming user browsers.) This specification/guidance

document will provide the requirements that an IETM provider must take into account when proposing or delivering such a capability for a JIA intranet.

- Common Browser. The immediate target for this specification will be the procurers of the user PEDDs (Portable Electronic Delivery Devices) and workstations, since the installation of this standard browser will be required for these devices. The browser software component must be pre-installed on the PEDD as is not included in the IETM View Package. However, the providers of the IETM will also be affected by this specification since their IETM must be formed/viewed using any JIA-compliant browser. Of course, a browser is necessary for acceptance of the IETM by the customer, since a usable IETM can not exist without a browser. Only two products dominate the Web-browser commercial marketplace, Microsoft Internet Explorer and Netscape Navigator, and the thrust and goal of this specification is to specify the configuration of each so that they will be functionally equivalent in any JIA intranet. This will involve some extensions to the commercially released products via specified plug-ins and controls; e.g., viewing capabilities common in Military IETMs, but not in the general marketplace, such as CGM (Computer Graphics Metafile) or the common PDF (Portable Document Format) used for legacy TMs.

- Electronic Addressing and Library Model. This is the overarching specification that holds the enterprise collection of IETM information together by means of digitally encoded and executable-link references. The specification itself will define the syntax and mechanism for building and executing the automated links to the IETM content and the IETM presentation software. Two additional areas regarding administration and enforcement of the recommendations are needed so that the enterprise-wide addressing concept will work. The Electronic Addressing and Library Model specification will discuss these aspects, which includes the actual bureaucratic administration and allocation of the DoD-wide IETM "address space". This is the actual indexing or URL-based electronically processable numbering system to which all the Services and their suppliers must subscribe. The specification will also discuss the important area of the library model or the search-and-access mechanism, which can be used to perform an intelligent content-based access to another IETM when the exact specific locator is not known. To support the proposed Library Search functionality, the specification will also specify and require metadata files, encoded in XML, which will serve as the primary search index files.

The Object Encapsulation and Component Interface, Intranet Server and Database Interface, Common Browser, and Electronic Addressing and Library Model performance specifications are all needed to effect interoperability of disparate IETMs in the field. The specific DoD form of these functional/performance specifications (i. e., whether they all should be formal DoD Performance Functional Specifications or some other type of guidance document) will be determined at the time the final recommendations are formulated at the end of the DoD Interoperability Project.

The overall interoperability goal is the ability to view any IETM with any browser that conforms to the JIA Common Browser Specification. It also requires that all cross references by one IETM to another IETM be encoded in a standard manner (i.e., in conformance with

the Electronic Addressing Specification) so that the IETM browser will be able to access the referenced IETM by a simple user selection (e.g., a mouse click). The other two specification areas are subordinate to these two user requirements, but are very important to assure that contractor-delivered IETM View Packages and the DoD infrastructure provide the needed user interoperability.

Th following paragraphs contain a short summary of the concept and philosophy embodied in each specification.

2.3.1 *Object Encapsulation and Component Interface Specification*

A core philosophy underlying this Architecture is that developers of IETMs can deliver, as a single View Package, all capability in the form of data and software contents needed to install and use an IETM on a standard DoD Intranet. This specification tells the IETM suppliers the form in which they are to package and deliver the digitally encoded IETM. This View Package may in fact contain both content data and software components that have been combined into encapsulated objects, and delivered as a contract package for electronic archiving or subsequent store-and-forward management. There is no provision for separately delivering an IETM player or piece of viewer software for other-process installation onto the user device. Rather the View Package will contain within it the capability to be automatically installable onto the end-user intranet at the time it arrives there.

The encapsulated data and software objects will eventually be delivered by the operational-Service Infrastructure to the field user activities as though they were simple binary data packages. These packages will be treated by the Infrastructure as file-oriented data destined for an agency Intranet Web Server. The packages will appear simply as a generic “bucket of sequenced bits” that make sense to the server but for which the content is of no concern to the infrastructure. The infrastructure activity need only make sure that these bits remain packaged together. The View Package is a set of industry standard binary files, each of which is assigned a JIA notional locator (e.g., a URL [Universal Resource Locator] conforming to the JIA addressing model) that contains sufficient information to support its installation as data in the Intranet Server file system.

The complexity and degree of integration of these View Packages will vary greatly among differing IETMs. Some will simply be a two-part collection of one software component and one data set. The simplest form will be a single set with all of the needed software contained in the standard JIA browser. In other forms, a system of software components and possible multiple data sets will spread out among several servers and the browser device when the IETM is operational. This would be the case when there are background software agents operating that might be performing diagnostics and system monitoring. Another emerging technology requiring complex IETM objects entails the use of software agents acting as an intelligent mentors inserting training aids into the job-aiding presentation when the agent (a computer program) determines they are needed. In-between there is a spectrum of complexity, each level requiring a different object-encapsulation approach. The “object” nature of these View Packages is that all the intelligence to construct the operational IETM on the target intranet is contained within the View Package objects themselves. There is no

standard for the internal constructs of the View Package in the JIA. This is the distinct characteristic of the object-oriented approach towards specification utilized by the JIA.

Until the point of receipt by the intranet server, the View Package is processed as a single object. There is a variety of mature approaches for bundling a set of files with headers into a single data set (e.g., Internet MIME [Multiservice Internal Mail Extensions] Standards). The Architecture may use any of them, requiring only that the View Package can be installed as a set of files on the intranet server(s). With this approach, no overt man-in-the-loop software-installation processes are required other than the automatic capability built into Web-capable browsers and servers. Many technical options exist for encapsulating View Packages; however, this requirement for automated software-component installation using Industry-standard Web practices is critical to determining the extent to which an encapsulation approach is satisfactory.

2.3.2 *Server and Data-Base Interface Specification*

The simplest way for the JIA to achieve IETM interoperability for the DoD is to utilize only generic Web-servers. Such an approach will not require that additional software be overtly installed on either the servers or the browser device. However, it is recognized that some legacy systems, and possibly some highly innovative new IETM applications, may require some sort of custom server extensions and database interface components. For complex IETMs, which require extended services operating on an intranet server, this installation of the IETM will likely involve two phases. One phase will be to extend the intranet by installing extended services, a process governed by the Server and Data Base Interface Specification, and the other will be to install the actual data and the required browser components, a process governed by this Object Encapsulation Specification.

Final recommendations on the use and encapsulation of server extensions will require additional technical investigations, since the technology and marketplace needs to mature before a full tradeoff and the development of specific recommendations can be accomplished. Some of the JIA support issues relating to these services are discussed later in this report.

2.3.3 *Browser Specification*

In line with the philosophy of this Architecture to use de facto Industry Standards, the browser requirements are established by the two particular commercial products, which together have captured essentially the entire Web-browser market. While it is possible to develop, assess, and evaluate a long list of needed and desirable requirements for the IETM browser, such an exercise would serve little purpose in light of economic and marketplace realities. New Web browsers are software products that are very complex and expensive to develop. Furthermore, the current products are being offered in the marketplace free of charge, effectively precluding the development of additional commercial general-purpose browser products. At this writing, these two products are Netscape Navigator version 4 and Microsoft Internet Explorer version 4. Except for a few, but very important, capabilities discussed below, these two products are functionally identical. For the traditional HTML 3.2 and earlier Web pages that dominate the WWW, they perform in a similar fashion.

The Browser Specification will specify the appropriate version of the two dominant commercial browser products and a set of standard extensions (i.e., controls and/or plug-ins) to these browsers. These extensions will include common DoD data viewers for file formats such as PDF, SGML (Standard Generalized Markup Language)/XML, CGM Version 4 Graphics, and CALS raster images. As an XML (Extensible Markup Language) capability will be in the basic functional set, the Version 5 release of these two products (expected by early 1999) will most likely be the baseline. These will be the first versions of both browsers to support XML and both providing companies (Microsoft and Netscape) have aggressive plans to add such a capability by that time. Mature Beta products supporting XML are available at the time this Report is being written. The inherent capabilities of the JIA-compliant browser will include basic presentation methods, either native to the commercial browser or added to meet JIA requirement, so that the component portion of the encapsulated object can be assumed to be preinstalled on the user device. In most cases, these particular components need not be included in the View Package. Native browser support includes components such as HTML layout, GIF (Graphics Interchange Format) viewers, and JPEG (Joint Photographic Experts Group) display.

One major area of difference between the two browsers lies in the area of object brokering and the automatic downloading of components. Ideally, it would be desirable to require that IETMs operate with either browser in its out-of-the-box form; however, the JIA Study team has concluded that such a policy would restrict some very needed capabilities regarding the downloadable components needed for the JIA object-encapsulation concept. The differences are due to the lack of cooperation on the part of the two competing companies, Netscape and Microsoft, to provide compatible solutions for the marketplace. The generic capability to automatically download and install of software components is essential to the JIA, so at the present time it may be necessary to chose one over the other for a specific implementation. To support users of Microsoft Windows 95, 98, or NT-based devices (which includes the vast majority of portable devices available), it is desirable to utilize products supporting the Microsoft DCOM object standards that provide turnkey support of this feature. For communities employing Microsoft software, it is strongly recommended that both browser products be enhanced (by third party plug-ins if necessary) to support DCOM objects (especially the downloadable ActiveX controls). These are the most efficient formats for executable programs running in a Microsoft 32-bit operating system

There is also a marked degree of difference in how the two products handle Dynamic HTML (DHTML), an emerging technology for putting intelligence into actual Web pages. However, because of the lack of consensus in the vendor community on DHTML standards and the fact that there are options for this functionality, the JIA Study team has not yet establish this requirement as part of the minimal baseline and discourages its use in DoD Programs.

The eventual goal is that all valid DoD IETMs be compatible with both the Internet Explorer and Netscape products. This may require some installed extensions to make the two products interchangeable to the maximum extent possible.

2.3.4 *Electronic Addressing and Library Model Specification*

The syntax for JIA electronic addressing will be based on the existing Universal Resource Locator (URL) standard for the World Wide Web because it is widely implemented in virtually all Web-enabled vendor products. Any occurrence of a legitimate URL string of characters is automatically made "hot" in the vendor application and a mouse click or two on that hot spot will launch a Web browser search which in turn will locate the file referenced by the URL and display it on the screen. In addition to requiring a standard syntax, the Electronic Addressing and Library Model Specification will also require that all of the Services maintain and publish a permanent registry of all valid references to the IETMs issued by that Service. Once published, a valid URL must not be changed. This type of URL is called a persistent URL. In order to assure that URLs are indeed persistent URLs, the JIA recommends the use of virtual URLs (vURLs). These are URLs that use an administratively assigned server reference, notated by URL syntax; however, the referenced server exists in name only. That is, it does not actually exist on a network and is used for data-management purposes only. When the IETM is actually installed on a network, the vURL is remapped to the actual server on which the IETM data resides. (This is easily accomplished in practice using what are called Host files and/or Domain Name Services (DNS) in accordance with standard World Wide Web practice.)

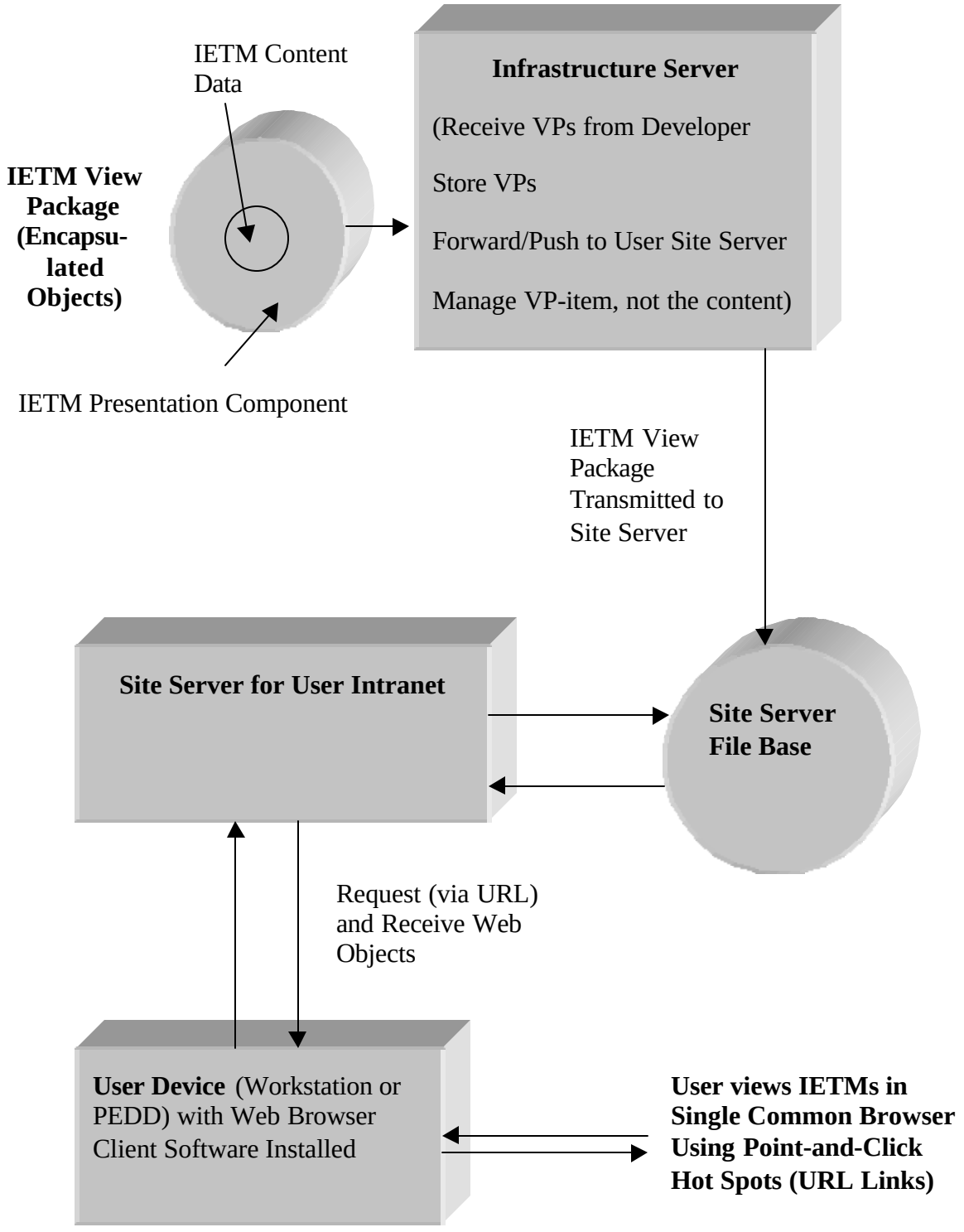
The specification will address the requirement for the remapping of these vURLs (which reference a hypothetical server on the World Wide Web) into the actual server and file-system locations on the intranet under use. The Specification will also outline a on-line search oriented Library Model and identify the requirement for a standard Metadata Package to support on-line searches. This metadata package will be encoded in XML and attached to each IETM View Package, which will contain indexing information useful for automated search engines in identifying an IETM reference as resolving an ad-hoc user searche.

3. BENEFITS OF EMPLOYING THE ARCHITECTURE

3.1 Basic JIA Operational Flow Diagram

Figure 1 shows the flow of IETMs through the JIA. It illustrates the employment of the JIA by the original IETM developer, the management infrastructure repository, the user-site intranet server, and the end user who selects the next object to view via a point-and-click Web-browser interface. The presentation components referred to can be either client or server components or implied (i.e., omitted) in the cases in which they are preinstalled in the standard browser.

Figure 1 – Flow of IETMs through the JIA



3.2 Benefits from the User Perspective

The principal benefit of the JIA is that the user can use a single personal device to read and utilize any DoD IETM, no matter which Service originated the IETM. To accomplish this, the end user accesses and views the IETMs with either a workstation personal computer in a shop environment or a PEDD (Portable Electronic Display Device). The portable device will be configured either as a network client attached to the unit intranet or it will be reconfigured to operate in stand-alone or detached mode. In either case, the display of the information on the user interface is identical, and the user cannot determine from the look-and-feel of a screen display the mode in which the device is operating.

A major benefit of the JIA to the user is that all information is viewed through a common and very familiar Web-browser interface. To access an IETM, the user will select an URL reference using one of the many access-screen or menu-select options available (e.g., favorites list, explicit entry, a pre-assembled list of active IETMs on a squadron Home Page, a hot-spotted index graphic, a Web-page job-assignment form listing the needed technical references). All of these are common practices borrowed directly from the World Wide Web community. From the user's perspective, the referenced IETM content simply appears in the browser window.

A major benefit to the user organization is that no explicit software installations are required to utilize an IETM even on a new out-of-the-box JIA conforming browser device. Depending on the browser security level set, the user may, at times, need to overtly accept software components that require installation by a single-click acknowledgment but for which no explicit installation action is needed because the browser installs the components automatically. This is an essential user-friendly feature of the JIA. Thus, there is no need for a trained and certified system administrator to install user software; that is a part of the simplicity of this approach and a tremendous cost saver.

Another key benefit of the JIA is that the Web-based access methodology is a proven "point and click" user interface. If one IETM contains a reference to another IETM, the user can click on the highlighted reference and that referenced IETM will appear in the browser window (assuming, of course, the referenced IETM exists on the user's intranet). This second IETM can in turn reference a third IETM. To return to the original IETM, the user simply uses the "back" arrow on the browser interface, effectively reversing the references. Modern Web browsers can handle many levels of such nested referencing with no performance degradation, a very powerful feature. From the user perspective, the JIA is thus intended to make the use of disparate IETMs as easy and "seamless" as possible with modern technology. Because of the nature of the Web-browser technology employed, the user experiences a great deal of common look-and-feel in the interactive (navigation-control) area, even if the individual IETM user-interface for the content varies.

A common practice on the World Wide Web is the use of search engines such as those employed by the well-known companies Yahoo and Excite. The JIA Library Model and

the required standard XML-encoded Metadata Package are specifically designed to facilitate the inclusion of search engines on a JIA-conforming intranet. In these search engines, the user will enter a list of key words or reference designators and the search engine will identify possible IETM references available on the user's intranet. The JIA will not specify the search engine, but a rich selection of commercially available search engines, which build their indices from XML- and HTML-encoded sources and can easily be employed on a JIA intranet, is expected. The ability to get all the information needed to perform a task in a timely and convenient manner has, from the beginning of the IETM concept, been one of the promised performance-enhancing capabilities of IETMs. This JIA implementation, using low cost commercially available technology, will permit realization of this promise.

3.3 Benefits for the IETM Developer

The principal emphasis of the JIA from the IETM-developer perspective is that all software components and data needed to make an IETM accessible on the JIA display device are bundled into a single digital product (i.e., the encapsulated object), which can be easily installed as a set of data files onto an intranet-server file system. The primary benefit to the IETM developer of that object-oriented concept is that the developer is free to choose whatever authoring and development environment he prefers. The JIA does not dictate how the IETM is to be developed nor what the internal format of the IETM object must be. The external interfaces are specified but they are in accordance with most of the modern electronic-document authoring environments which are rapidly being adapted to operate on the World Wide Web and, as such, should operate equally well on a JIA-compliant intranet. Proofing tools for the IETM objects are also easy to set up as a JIA-compliant intranet and the JIA browsers are made up of readily available software products, which the authoring activity can easily procure without going through DoD supply channels. The design philosophy for the JIA is to use the best readily available commercial practices for developing and deploying IETM products. The only providers that might not benefit from this approach are those which have a lock-on to provide high-cost and non-interoperable "stovepipe" solutions to a single DoD program. The JIA has been specifically developed to discourage such costly (or at least costly in the long run) practices.

While the technology needed to bundle all of the IETM components into a single digital package is complex, it is readily available in COTS products. These are very inexpensive (relative to traditional IETM products) and easy to obtain in almost any mass-market computer software store. This positive situation has resulted form the exploding popularity of the Internet and the World Wide Web for commercial applications. As a result, there has been an unprecedented rush by suppliers to get competitive products into the market. A foundational principal of the JIA is that the products developed for the Internet can be used unmodified to develop IETM products for a JIA-compliant Intranet. This process is in sharp contrast to a conventional IETM application where the IETM product is delivered as two separate items, the IETM content data and the IETM presentation-system software program.

3.4 Benefits to the DoD IETM Distribution Infrastructure

The primary benefit of the JIA to the DoD IETM Digital Distribution Infrastructure is that encapsulated IETM objects can be distributed without requiring that the distributing system know what is inside the electronic capsules. The Infrastructure activities can therefore be simple distribution centers, for which the DoD has substantial experience, and not data-processing centers, which the Government would find much more difficult to operate and staff.

Within the JIA, the complete set of IETM data and the associated components is called a View Package. All data and component delivery to the end user is accomplished through the Web-based client-server interaction. A feature of this concept is that the View Package can be passed, unmodified, from server to server as part of the JIA electronic-distribution system. The key JIA concept for creation and use of the Infrastructure Server is that the IETM View Packages are composed of self-contained digital objects that appear to the infrastructure as large standard binary formatted digital files. These objects can be received from a developer, stored, forwarded, and delivered from one server to another without any need for the infrastructure agents to know the internal structure of the View Package itself. The infrastructure site can function more as a supply center than as an information-systems center.

The specific design of and development of a specific DoD component Infrastructure was not in the scope of this reported JIA Effort. Such an infrastructure design will undoubtedly be a complex, difficult, but important task that will be complicated by the impact it will have on many existing business practices. However, this key JIA element, which enables the objects to be processed as an item of supply (with no requirement to manage the internal content or structure of the object), should make this task much more manageable.

4. ARCHITECTURE TYPES

In practice, the implementation of an IETM intranet may be simpler (as is the case with basic HTML pages) or more complex (as is the case with most custom servers) than the baseline Operational Flow described in the previous section. The following breakdown of anticipated IETM View Packages by Architecture Type is presented at this point in this Report in order to categorize those variants and to provide guidance that is more specific for implementation of these variants. Any particular IETM intranet implementation will typically contain a mixture of these Types. The four Type categories represent a continuous spectrum of variation (i.e., some applications will be difficult to categorize precisely). However, identifying the Types will make it easier to present guidance in the JIA, particularly in regards to the Server and Data Base Interface Specification. Definitions of these Architecture Types are given in Table 1.

Table 1 - Proposed IETM Architecture Types

Type	Characteristics	Examples
Type C1: Basic HTML/ XML Pages	HTML/ XML page(s) with only browser-resident components. Requires no component licensing. Most will work on any browser. Includes HTML 4.0 scripts. Client processing only. "Plain vanilla" HTTP server.	HTML with Java script, GIF, JPEG, frames XML + XSL Style Sheets
	Note: HTML/ XML pages may be used to include one or more Type C2 custom components. If the HTML/XML tags no displayable content, it is considered Type C2. If it does contain tagged data, it is a combination C1/C2. .	C1/C2 examples: HTML file plus Java bean(s) HTML file plus plug-in HTML file plus ActiveX control(s)
Type C2: Simple Component	One data set plus one custom automatically downloadable non-HTML component	.pdf plus Acrobat reader control .doc plus WordView control
	May be nested Type C2 data-set/component pairs ("encapsulated objects"). First component loaded into browser shell/container has capability to access another client component and associated data set under control of original component. Requires component licensing. Not recommended for new applications. Client processing only. Uses "plain vanilla" HTTP server.	Legacy Systems reprogrammed as custom browser or presentation system operating inside a standard browser shell/container.
Type S1: HTML Plus Application Server	Two-tier architecture in which Web page includes reference to server application(s), which must operate before page, is delivered to client as Type 2 HTML/ XML. Data and components managed on server). May utilize database co-located on server but most content is in web page files. Requires HTTP server with components for server-side computations. Requires client *and* server processing.	MS Front Page Webs MS Design-time Controls CGI Server Apps DynaWeb
Type S2: HTML with Database Server	Three-tier architecture that includes a Web page server with pages functioning like a template; e.g., for calls to a database, which contains most of the IETM content. Can include server and components for custom functions. Requires a database server (e.g., Oracle) in addition to the HTTP server. Can use MIL-PRF-87269 Data model for data base on DB server. Permits both Client *and S*erver processing.	AIMSS 4.0 (planned) GD TechSight Web MS ASP w/ODBC Calls

These Type definitions are grouped into two categories:

(1) The baseline architecture, IETM Architecture Types C1 and C2. Definition of these two client-centric Architecture Types has essentially been completed. These Types require only a browser and a generic HTTP server.

(2) The extended architecture, Architecture Types S1 and S2. For these server-centric Types, the technology for employing the additional servers in the Web environment is less mature and more diverse. This segment of the market place is just now emerging, and it is still dominated by proprietary products. (This situation is in large part due to the fact that vendors have opened the browser products to the public domain, a market in which there is little money to be made, and have kept the server market proprietary, where the vendors see profits to be obtained and seek competitive advantage.)

4.1 Characteristics of Architecture Types

Architecture Types C1 and C2 share common important characteristics in that they do not require installation or operation of unique software on the server. Thus, the server can be treated as an electronic bookshelf. As far as the server is concerned, the two parts of an encapsulated object (the data and the associated software components) are simply treated as files. Additionally, any contemporary HTTP server can be employed and it does not matter what operating system is employed. Thus, for Type C1 and C2 IETM applications, interoperability is very low-risk in the sense that, with these, any IETM View Package can be accessed using any server. For the server requirement for Types C1 and C2, only a generic server is required and no JIA-specific server specification is required. Both Types are considered pure encapsulated-object Types; however, for Type C1, the component part of the object can be implied (i.e., omitted), as its presence can be assumed as preinstalled on any JIA-compliant browser and need not be included in the transported IETM View Package.

The Type C1 definition is closely tied to the specific versions of HTML and XML, a factor which needs further clarification in this document. For planning purposes, it is recommended that foreseeable emerging standards (and not current standards) for both HTML and XML be used to define the JIA requirement. In this light, HTML/XML is herein specified as employing both HTML version 4.0 and XML version 1.0 (including the associated XSL style and XLL linking specifications), when these two International W3C (World Wide Web Consortium) standards are formally approved. HTML 4.0 is a mature specification and should soon be approved, while, the XML family of standards is still a year or two away. There are several reasons for this recommendation. The future standards will almost certainly be relevant in the time frame when most applications are developed according to the proposed Architecture, so the best estimate as to what will be applicable should be used. Vendor implementations of the draft standards are available now for test purposes. Another important consideration is that there is written commitment of essentially all the major software vendors to support the future standards, whereas there is no complete agreement on the delivered-product support of the current

standard (i.e., HTML 3.2). In particular, vendors have indicated support of the emerging HTML 4.0 standard. Additionally the XML standard has elicited widespread vendor promise of support as a user-extensible expansion of HTML. XML lags behind HTML 4.0 in maturity, but is sufficiently complete so that prototype software exists from major vendors, and shows promise of becoming a Web-based tagging standard that is more suited to complex IETMs than HTML. In particular, it will be much easier to convert the large DoD inventory of SGML-tagged source data to XML for a run-time object than it is to convert it to HTML for presentation.

For Type S1 and S2 IETM applications, particularly for the server application (i.e., software), the situation for ascertaining de facto industry practices is much more complex. Several approaches are available for standardizing many of the issues such as Microsoft's design-time controls, Active Server Pages (ASP), and Front Page server extensions, and a variety of third-party middle-ware products; but they are all proprietary and not universally accepted. The technology and state-of-COTS are not sufficiently mature at this time to propose any one of them as a DoD standard so that all IETMs can operate on a single server. To achieve operational interoperability with a particular server in the short term, there are two possible approaches for a working solution:

(1) The various IETM providers must put their own physical server(s) plus the IETM View Packages on the shared user intranet (very feasible with the state-of-the-art and capacity of today's portable computers and plug-in network standards); or

(2) Require that all IETM creators use the same sets of server components and that the standard components be installed on all intranets employed in the community throughout which the IETMs are interoperable. However, for ad-hoc Multi-Service operations, this alternative is not considered feasible.

4.2 Elements Diagrams for Architecture Types

The core Architecture (Types C1 and C2) requires two kinds of software elements: client browsers and Web servers, as illustrated in Fig. 2. In general these are hosted on separate devices connected by a TCP/IP network (i.e., LAN); however, an intranet can also be set up in a single display device without a network. In the case of IETM Architecture Types C1 and C2, these two kinds of elements are all that is needed. In the case of Type S1 (see Fig. 3), a requirement exists for an additional element, the application server, sometimes referred to as a Web-server extension, since it effectively operates in the same operating system as, and is an extension of, the HTTP server. In the case of Architecture Type S2 (see Fig. 4), there is the additional requirement for a Database Server which hosts most of the IETM content, which may or may not be hosted in the same device as the Web server. A Type S2 application usually include aspects of a Type S1, since it requires an application server to process the data-access and request dialog between the Web server and the separate database server. Note that while the line between these two Types may, at times, not be clear, in general they differ in where the primary data content is stored; i.e., in the server files or database server.

Figure 2 - Elements for Architecture Types C1 and C2

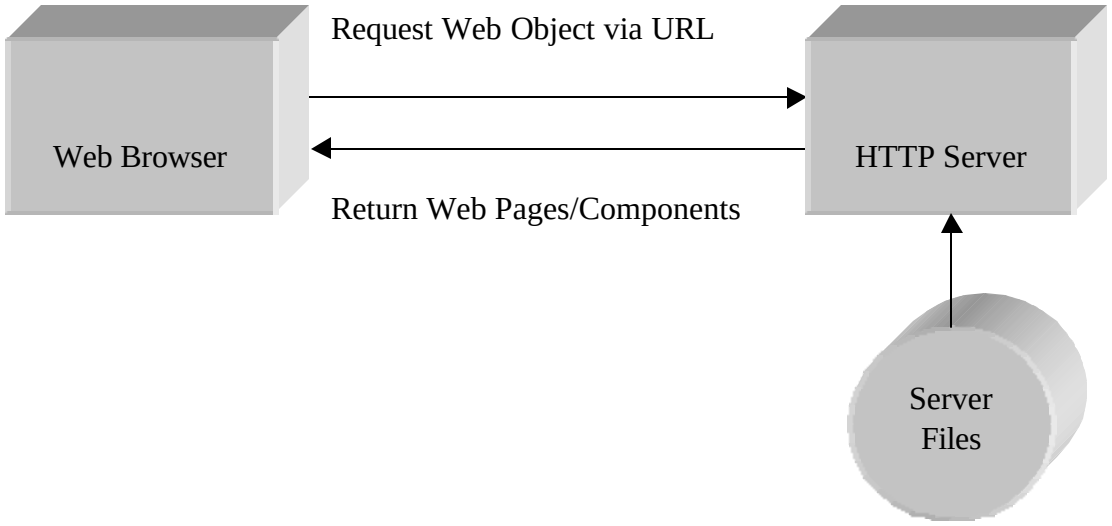


Figure 3 - Elements for Architecture Type S1

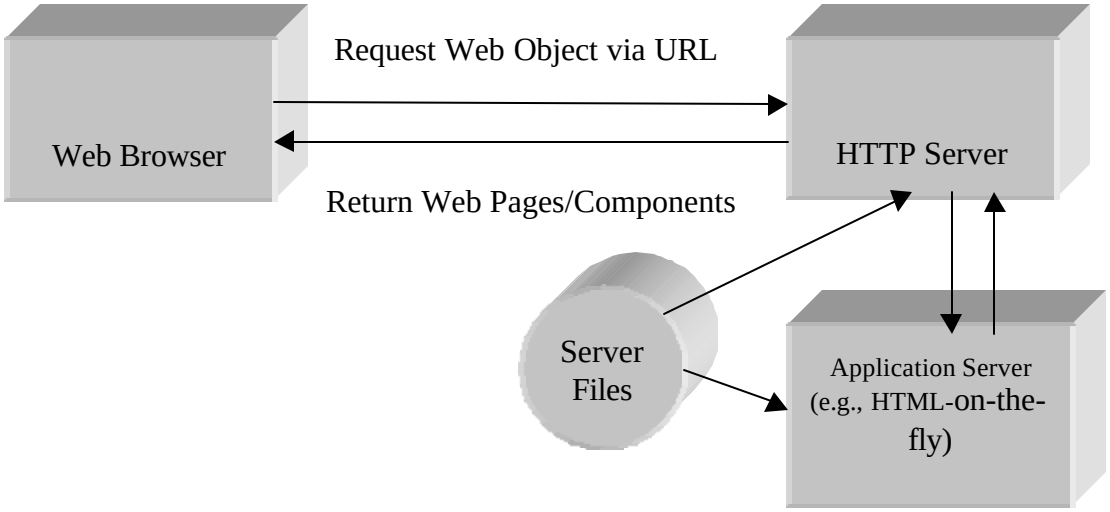
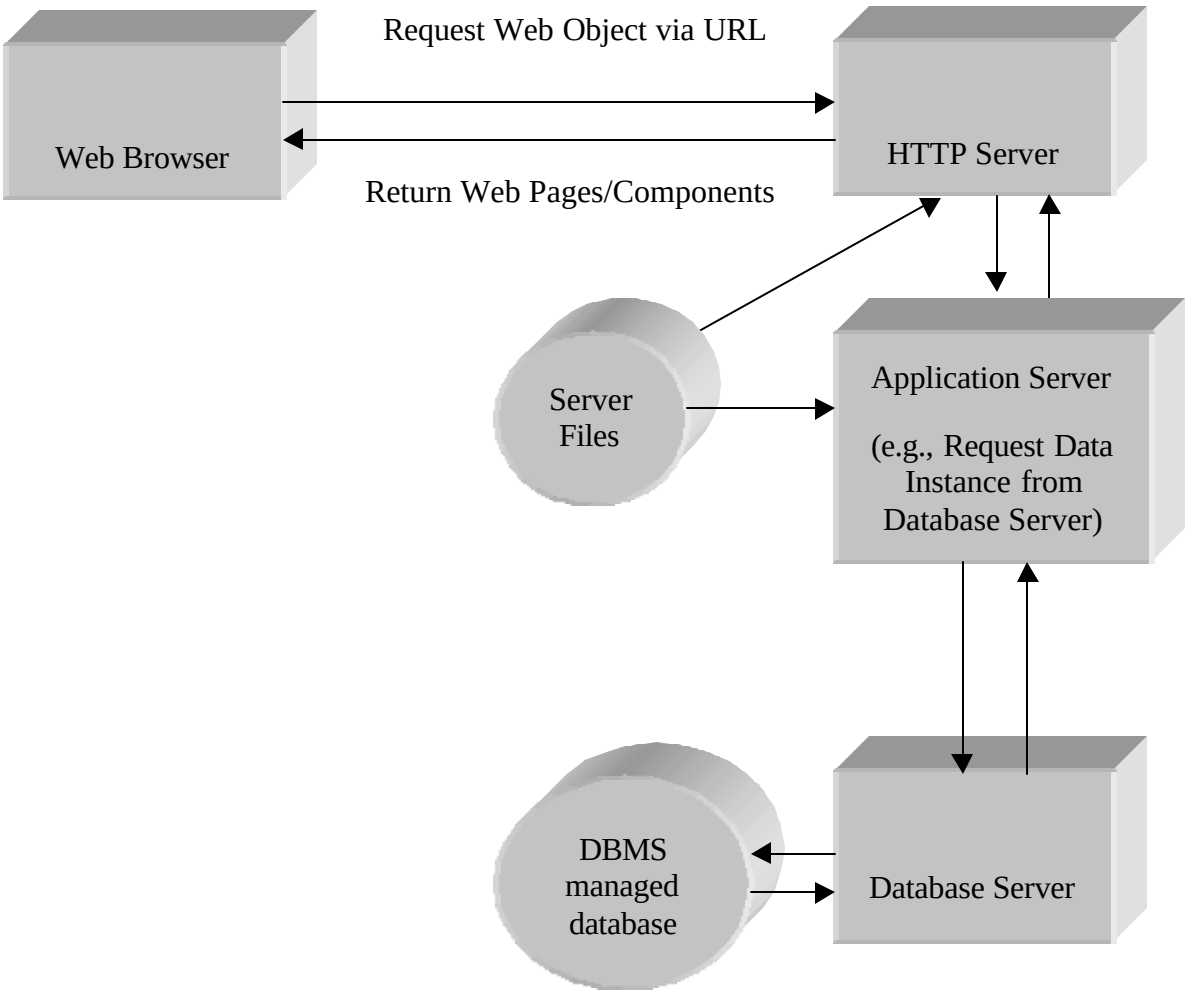


Figure 4 - Elements for Architecture Type S2



5. EXPECTED PORTABLE ELECTRONIC DATA DEVICE (PEDD) ENVIRONMENT

A unique feature expected of a typical JIA Intranet, as opposed to more conventional intranets, is that the common mode for the PEDD (or other display device) is to operate as a stand-alone device. Portable devices are likely to be disconnected from any network during the time when an IETM is actually being viewed in support of a maintenance task. The PEDD is connected to the intranet network only occasionally for purposes of downloading new or updated information and the uploading or reporting of information.

The NSWCCD laboratory has shown that it is possible to bring all the functionality of a distributed intranet to a single device by installing a personal Web server on the PEDD and, to the extent needed, all other servers which might be needed for Architecture Types S1 and S2. This is not difficult to do in practice; especially with the widely used Microsoft NT Workstation operating system, which includes a personal Web server. For Architecture Type S2 IETM applications, when the PEDD is used in stand-alone mode, there will be a need to explicitly install the database management system (DBMS) which performs the database-server function.

A need also exists to perform server-request redirection by which all server requests in URLs are redirected back onto the PEDD server file system. There are several off-the-shelf approaches which accomplish this function (e.g., Windows HOST file, Local DNS, etc.) and all will implement the Architecture. As actual applications are developed, a substantial requirement will emerge, of course, for configuration-management facilities to be built into the downloading system that is supplying data to the PEDDs. However, with these self-contained intranet features in place, it will be possible to access any object loaded onto the device in exactly the same fashion as from the site server.

There are two options for running IETMs in the stand-alone environment which do not require the use of a web server on the stand-alone device, and which do not really conflict with the JIA concept. Individual Services may sponsor dual use implementations of these IETMs utilizing the stand-alone version for primary Service use, while at the same time maintaining the option to mount the IETM on a JIA-compliant intranet (and hence become a JIA-compliant IETM) with little or no additional effort. These options do not apply to all IETMs and will not work for IETMs that require a server component (i.e., Type S1 and S2 implementations). One option takes advantage of the fact that both the Netscape and Microsoft Web browsers can directly access a file system on a local computer without using a server (including a CD/ROM mounted on the computer's file system). These applications are commonly called disk webs, and are used by book publishers to distribute CD/ROM versions of their publications, an efficient alternative to employing PDF. This approach is, in general, limited to static presentations such as book replicas. If the disk web limits its internal URL references to a restricted syntax called "relative addressing", it is possible to mount the same IETM system on a JIA-compliant server or on a local computer's file system. The other option involves a legacy-data implementation and format for which a JIA-conforming Web-enabled presentation component has been developed that requires no alterations of the original electronic

information for presentation on an intranet. In such a case, the information can still be viewed on the original stand-alone viewer.

6. JIA SUPPORT OF APPLICATION SERVERS AND DATABASE MANAGEMENT SYSTEMS

Minimum server capabilities are highly dependent on the type of Architecture of the system being utilized. For Types C1 and C2, virtually any commercial HTTP server can be utilized. In these cases, only the Object Encapsulation and Software Component Interface specification is needed to define the requirement to IETM suppliers for the packaging of their IETM View Packages. The ease of installing Type C IETMs onto a Web-server makes they the preferred Type for support of Joint Operations when an ad-hoc electronic library needs to be quickly assembled.

However, for Type S1 and Type S2 applications, JIA support is neither straightforward nor easily standardized. The initial JIA specifications for these servers will be in the nature of guidance documents. The JIA supports these Type S applications because they provide a level of system functionality and power which is very desirable in the support of very large IETM installations. While it might be possible from a standardization viewpoint, simply to forbid the use of these "back-end" servers, the reality of best commercial practice is that these types of products provide powerful and needed functionality. Often they cannot be totally ignored by Program Managers seeking to maximize the use of computer technology to solve real problems and must be accommodated in the JIA.

For these Type S applications, the developer of the IETM will have to provide software and data components that need to be installed on one or more "back-end" servers, i.e., the application servers and/or the data base servers needed for Type S IETM applications. The concept of auto-install that is specified for the browser components will not, in general, apply to the administration of these servers. While it is the goal of the JIA that browser devices will require no system administration, the servers will still require an administrator for the installation of server software components. Thus, for a Type S Architecture category, the provider of the IETM will have to arrange separately for the installation of server components on the target intranet (governed by the guidance contained in the Server and Database Interface Specification). This typically one-time installation will be in addition to providing the actual data and browser components as an encapsulated View Package (governed by the Object Encapsulation and Software Component Interface Specification).

The ability to provide tight guidance in the JIA regarding back-end servers is complicated by the state of the technology that is continually emerging and evolving in the server marketplace today. There is economic pressure on software vendors to develop a competitive advantage (i.e., proprietary features) in their server products, since it is widely recognized that profits will be made only in the server marketplace, as opposed to the browser marketplace. As long as Microsoft and Netscape continue to make their powerful browser products available free of cost, there is no money to be made in the browser market. Thus, vendors will seek to make their profit in the Server market, and a

direct result will be a proliferation of proprietary server products, a situation that will continue to complicate the standardization of DoD servers.

Specific considerations for presenting guidance for the back-end server capability in a JIA-compliant intranet are summarized below.

6.1 Type C Server Support

Virtually any robust commercially available server product running on any operating system will support Type C applications since all intelligent processing is performed in the browser and not on the server. All the server needs to do is serve out HTTP referenced files and Web Pages.

6.2 Type S Server Support

The Type S Server Support requirement is a function of the Type S Architecture Type itself. Several varieties of Type S applications will require extensions beyond a simple HTTP or Web server. One approach is to use a proprietary server which, when it is installed, automatically provides specific custom application software. Two examples of this type relevant to IETMs are the DynaWeb product, a Web-enabled version of the popular Dynatext electronic text viewer, and the TechSight Web product of General Dynamics. These products offer powerful functionality because of the server software, but have the distinctive problem of creating a life-cycle software maintenance requirement for the life of the product, since the software must be modified to upgrade the functionality of the IETM; that is, they do not have the commodity character of an out-of-the-box product such as the Microsoft IIS (Internet Information Server). Another variety of a Type S server extension is closer to this commodity situation and involves a standard set of server extensions such as Microsoft's Active Server Page (ASP) extensions to IIS, which are actually installed as a software extension the Microsoft IIS Web server. Functionality is added to the server by means of server components that are included in the IETM Web-Page which are automatically installed on the server in a manner similar to the automatic component installation for the browser, as in the case of ActiveX controls. Microsoft does offer such components and the tool kits to develop them in its Visual Studio product (especially Visual InterDev) and a meaningful level of support in its low cost Front Page 98 product. They are, however, proprietary to Microsoft.

6.3 Type S2 Server Support (Database Interface)

Type S2 Architecture applications are a particularization of the general Type S application. They are server applications in which the content data is primarily resident in a DBMS-managed database and the object encapsulation in the file-based web pages serves as organizing shells or templates. In fact, in an IETM for which the format has stabilized, there may be no need to modify the portions of the encapsulated objects managed by the Web and Application servers when content changes are made. Only the database instance needs to be modified. Such a construct was envisioned when the

"Class 4 IETM" was prototyped almost ten years ago. Virtually all database vendors are marketing a Web-enabled variant of their DBMS. This is an emerging area in which new products are being developed every month, many of which are applicable to IETMs. Thus, this is not the time to restrict or standardize the Type S2 solutions since affordable COTS technology is now emerging in the marketplace.

A particular impact of the utilization of Type S applications is that they will need to support the occasionally connected scenario in which a PEDD has to be downloaded with an IETM and then detached from the intranet in order to perform work at the maintenance site. This procedure requires that a local copy of the application and/or database servers be installed on the portable device and that there be some facility to keep the local copy of the database synchronized with the main database on the intranet. This requirement is not unique to DoD application and commercial products exist to perform this function. The Specification will specifically provide guidance for this situation. As is typically the situation with these Type S applications, the available solutions, while powerful, are proprietary and not amenable to standardization. Despite this lack of standardization, these COTS products can be deployed on a JIA intranet, largely because they are being developed for commercial application on the World Wide Web.

The approaches outlined in this Report address general technology and object standards; they do not provide an assessment of individual commercial products. However, a vendor trend that is only recently emerging is to introduce attractive features in proprietary intranet-oriented server products especially in the use of DBMS products. The original concept, upon which the initial NIA Effort was based in October 1996, was to allow any number of proprietary authoring products and associated methods to be encapsulated into the IETM objects, but to utilize only generic servers in the field. The subsequent DoD project is still holding to this approach, but during future study phases, it must investigate selected server-extension products that solve problems not easily solved by pure encapsulated objects. In particular, this situation occurs when a Type S2 IETM application requires the services of a separate DBMS as well as the presentation method that is encapsulated in the IETM object. In this case, it will not be feasible to force that DBMS into the encapsulated object. Other commercial server extensions will be examined in future studies, but, in general, these approaches risk introduction of a high infrastructure cost and may create a situation involving proliferation of non-standard servers that is not in the interest of interoperability.

The Specification being developed for Server and Database Interfaces will actually be a general guidance document and one that is expected to evolve with the technology.

7. JIA SUPPORT TO ELECTRONIC ADDRESSING AND THE LIBRARY MODEL

Implementation of the electronic addressing function requires the following:

(1) A mechanism and format (i.e., syntax) for encoding electronic addresses into an IETM View Package;

(2) A defined name space and address model (i.e., electronic numbering system); and

(3) Searchable index information for each IETM entry point that adheres to a established convention so that intelligent search engines can locate an IETM reference when the specific locator address is not known.

The third area is characterized by what is called “metadata”, or a set of searchable keyword and other identifying information in a standard format that is associated with each IETM and can be used by a search engine to identify sought references.

For use of the JIA, it is recommended that the Internet URL (Universal Resource Locator) be the format for the address reference syntax and the Industry standard practice of employing URL references for automated electronic linking be adopted. This practice makes URL references embedded in an electronic document visible as hot spots which cause the default browser to open up a new page with the content of the referenced URL displayed in that page. Thus actual URL coding can be hidden under a more human-readable text string or graphic, as in the case of most Web-page authoring programs; or the URL itself can be made the hot spot as is the case with MS Office Products, such as MS Word 97. Additionally, the Electronic Addressing practice recommended will employ virtual URLs or vURLs that, once established, remain the same no matter where, or on which intranet or server, the object resides. These virtual URLs will contain a server reference for accounting and configuration-management purposes, but the server so named will not, in general, actually exist on any real network. The intranet can remap the virtual server references in the vURL to the actual server site on which the files exist using standard features such Domain Name Service (DNS) or other server-names-to-actual-IP-address mechanisms such as the Windows or UNIX HOST file. In this light, the server name required does not need to exist on an intranet because it will always be remapped onto an actual intranet server when operational. Guidance for establishment and use of vURLs is documented in Figure 5.

Figure 5 - Guidance for Establishing vURLs in the JIA

vURLs will be authored and maintained as follows:

HTTP shall be the Web-page protocol to be utilized in this Architecture (i.e., the URL starts with “HTTP://”)

Server Name is to be in the form of “natsf.navy.mil” and is listed as though it were an actual server on INTERNET. It is recommended that management activities actually install such a server and maintain all of their cognizant URL references on that site in the form of acknowledgment as valid reference even if the actual content is not included for security reasons until such time that a secure DoD network is available.

File/Directory notation is to be unique across all DoD IETMs and is administratively assigned as though it were the IETM number in the form of a Unix file system reference with forward slashes such as "/navy/f18/ef/engine/ge/"

Additional Directory breakdown of files within the IETM reference is merely a further extension of the Assigned File/Directory name and content for a section within an IETM and may be null for the top level reference.

Sample: "/navy/f18/ef/engine/ge/diagnosis/test3".

Optional user-defined moniker may be utilized. These are most commonly used for carrying detailed database access parameters in the URL.

Regarding the actual addressing model, the JIA will require only that such a model exists and that it uses the URL syntax. However, the actual administrative task of establishing, assigning, and enforcing the administration of the address space will have to be the responsibility of some standing management activity and not of the JIA Technical Team which is developing the JIA performance specifications, although the Technical Team can provide a number of suggestions.

Establishing the area-library model or searchable-access mechanism is another area in which the most difficult task will not be primarily technical. The challenge for an administrative activity will be in securing DoD wide agreement for adopting a standard format for the searchable metadata. A JIA implementation can operate without metadata by relying on hard-coded URLs for linking the Web and IETM information together. However, the real world experience of the World Wide Web, has shown that in practice, users rely extensively on search engines to locate needed reference information that the content providers did not foresee when the viewed content was authored. There are two basic approaches to developing a searchable data base of such metadata: one is to create it after the fact by utilizing a semi-automated third-party indexing service or mechanism (the approach commonly used on the World Wide Web); the other is to require that the content providers to author a searchable information data set in a structured format. The JIA technical team is investigating the later approach, with the most likely recommendation being the use of an ASCII encoded XML data set employing a standard set of XML tags.

8. ASSURING AND TESTING COMPLIANCE

The requirements of this Joint IETM Architecture will be specific when applied to a particular intranet implementation, and it is possible to prepare a checklist and list of criteria to determine whether the requirements have been met in any given case. Accordingly, it is recommended that for any specific operational community, the primary compliance test requirement be based on establishing whether IETMs generated using the Architecture will work in the particular operational intranet environment being utilized

by that community. To the extent that the Service Component utilizes commercial environments, such testing is easily replicable on PC's in the IETM producer's facility or at the Infrastructure receiving site. Because the Services will be required to utilize the JTA in developing these operational intranet environments, the obstacles to the interoperability with other DoD intranets should be minimized.

A more difficult, but equally important, factor requiring demonstration of compliance involves the electronic-link references. There are two aspects to this question:

(1) With regard to the validity of the linkage, it must be established that the specific URL matches exist once the basic referenced document has been installed in the Intranet and, when a custom component has been automatically installed in the browser device, that the linkage methodology is working properly.

(2) There is also a need to create a usage guide for the URL references, which has not been fully developed at this time. This need is an administrative requirement regarding the use of correct reference content, not an technical issue.

Certain easily avoidable practices will operate in the Internet but will not work well in a remappable JIA-compliant intranet that uses vURLs. For example, the use of fixed (i.e., numeric) Internet IP addresses and the use of absolute instead of relative internal URL references in an IETM must be avoided. The associated acceptance and compliance demonstration procedures must also be developed. Implementation of such a capability will be a major future task for the entire DoD Community and a necessary part of developing a capability for acceptance testing of IETMs submitted by weapon-system contractors.

9. MIGRATION AND INTEGRATION OF ELECTRONIC LEGACY SYSTEMS

This Architecture has intentionally been designed to accommodate legacy systems with no modification of the legacy data format. However, establishment of this capability will require that the legacy presentation software be converted to a Web-compatible software presentation component for those legacy systems that will not be replaced by an alternative data format. For many legacy-data formats, the needed Web-capable components have already been developed. Examples are PDF, MS/WORD files, and most common graphic formats. However, some custom DoD IETM-presentation systems have not been converted. In these situations, the current application would have to be converted into a form compatible with the Web browser such as an Active-X control or a Java-Beans application. (See attached Glossary for an explanation of these terms.) More complex applications, such as those utilizing a DBMS, need to be converted to a Web-compatible system for performing functions such as database access. The conversion effort is typically more difficult for an application that is programmed in an older 8- or 16-bit application programming language. Newer applications using 32-bit development tools, especially those developed for Windows platforms, will experience much less software-conversion effort. This is because ActiveX is based on the earlier OLE (Object Linking and Embedding) standard used in most Microsoft Windows-targeted software applications. In other applications for which there is a large but not growing inventory of

legacy material, it may be more appropriate to perform a one-time conversion of the data to a format more amenable to standard Web presentation. Such issues must be evaluated on a case-by-case basis.

10. IMPORTANT UNRESOLVED ISSUES NEEDING FURTHER STUDY

A variety of technical issues and implementation issues within the original scope of this Study remain for which the JIA Study Team is not prepared to make formal recommendations without further study and evaluation. In many cases, more time is needed for the completion of Industry standards and the establishment of de facto standards in commercial practice, and for the emergence of more mature vendor products. However, these details will be needed for a complete architecture and are cited here only for information. Two essential areas are discussed below. Additionally, there are many related management and technical issues outside the scope of this Effort, which will need attention in the future. These involve areas such as the configuration management of the IETM View Packages in many distributed data repositories (i.e., intranets) and the development of an electronic library model with sufficiently comprehensive indices and other metadata files to facilitate specific access to the correct IETMs as the installed inventory gets very large.

10.1 Maintaining a Common Look-and-Feel among Differing IETMs

While the use of the common browser does standardize many of the user-interaction features, it is very possible to include a custom component that contains its own set of unique user-interaction features layered under the higher-level browser toolbars. These features often conform to a proprietary look-and-feel standard. A well-known example is the PDF Acrobat Viewer component, which includes all the Acrobat user features within the browser viewing window for features such as zoom, next page, scroll, etc.

A requirement still exists for a procurement-guidance document for minimizing the differences in look-and-feel among various disparate IETM presentation components that operate in the JIA environment. From both the Training and the Job Performance perspective, the effectiveness of each product is enhanced when it is displayed in accordance with a standard style, even if the actual underlying IETM presentation components vary and are proprietary in nature. The specific recommendation of this study is to revise the existing MIL-PRF-87268[1] specification to apply to the JIA framework and to make that revised specification available to IETM procurement officials as an acquisition tool. The revision philosophy should be to greatly reduce the existing performance requirements to those few that are really needed, and to tighten down those few remaining requirements to be as specific as possible. IETM TMCRs (Technical Manual Contract Requirements) and other procurement instruments could then require that delivered IETM View Packages conform to both the JIA performance specifications and the revised MIL-PRF-87268 user-interface requirement for a common

[1] MIL-PRF-87268 ***Manuals, Interactive Electronic Technical: General Content, Style, Format, and User-Interaction Requirements***. 1995.

DoD IETM look-and-feel interface. The real challenge in this proposed Effort will be in achieving consensus among the Services regarding the selection from competing recommendations regarding the implementation of control buttons, layout rules, etc. These are not technical issues.

10.2 Updating DBMS-Managed Data on a Server

The Type S2 Architecture application is the most likely mature architecture for traditional Class 4 IETMs (i.e., those based on MIL-PRF-87269[2]-compliant databases); however, the technology and products to support these Web-oriented database applications are currently immature, and definitive products that are still emerging in the marketplace. An area particularly needing continuing assessment regarding its role in the JIA is the updating and synchronization of databases in the field. In practice, the least risky way to update such databases is to use the tools applicable to the particular Data Base Management System (DBMS) being utilized. Most DBMS vendors have very good proprietary data-replication facilities for this very purpose. While these are network-enabled, the data-replication facilities typically utilize network protocols and procedures different from those peculiar to the World Wide Web. However, there is evidence of a strong Industry trend to blend these two technologies (Database and Web technology), and a high likelihood that de-facto industry practices will arise in the near future which should be applicable to the future JIA recommendations.

11. BUILDING AN INTEGRATED PRODUCT SUPPORT DATABASE

This Report closes with the following short but significant recommendation. It applies to the applicability of the JIA model to fielded weapon-system-support applications other than IETMs, such as job-site training, diagnostics, and logistics support. The Joint IETM Architecture can apply to any of the components of an Integrated Product Support Database (IPSDB), including training products used to support a weapon system in the field, on-line parts-ordering functions and parts information, and remotely operated diagnostics procedures traditionally known as test program sets. In developing integrated support for a weapon system, it should be the DoD position to discourage the development of monolithic support products for individual weapon-system support functions. Instead, it is recommended that a strategy be developed for using the proposed unified IETM Architecture to provide IPSDB functionality incorporating fielded technical training, diagnostics, and logistic support products. The family of general-purpose commercial products being developed utilizing Internet World Wide Web technology can provide all the functionality needed in these applications. These can be adopted instead of the traditional Stove-piped application software traditionally employed to develop custom DoD product-support systems.

[2] MIL-PRF-87269 ***Data Base, Revisable: Interactive Electronic Technical Manuals, For the Support of***. 1995.

12. ABBREVIATIONS

AME	Automated Maintenance Environment
ASP	Active Server Page
CGI	Common Gateway Interface[3]
CORBA	Common Object Request Broker Architecture[4]
COTS	Commercial-Off-The-Shelf
DNS	Domain Name Service[5]
DCOM	Distributed Component Object Model[6]
GCSS	Global Combat Support System
GIF	Graphics Interchange Format[7]
HTML	HyperText Markup Language
IETMTWG	IETM Technology Working Group
IIS	Internet Information Server[8]
JCG-CE	Joint Commanders Group for Communications and Electronics
JLC	Joint Logistics Commanders
JPEG	Joint Photographic Experts Group[9]

[3] For performance considerations on using the CGI, see Internet URL at: http://www.pcwebopaedia.com/CGI.htm

[4] Short for Common Object Request Broker Architecture, an architecture that enables pieces of programs, called objects, to communicate with one another regardless of the programming language they were written in or what operating system they run on. CORBA was developed by an industry consortium known as the Object Management Group (OMG). See Internet URL: http://www.pcwebopedia.com/CORBA.htm

[5] DNS is an Internet service that translates domain names into IP (Internet Protocol) addresses. See Internet URL: http://www.pcwebopedia.com/DNS.htm for a list of Internet Resource dealing with DNS.

[6] Short for Distributed Component Object Model, an extension of the Component Object Model (COM) to support objects distributed across a network. DCOM was developed by Microsoft and has been submitted to the IETF (Internet Engineering Task Force) as a draft standard. Since 1996, it has been part of Windows NT, and is also available for Windows 95. See Internet URL: http://www.pcwebopedia.com/DCOM.htm

[7] Technically, a GIF uses the 2D raster data type, is encoded in binary, and uses LZW compression. There are two versions of the format, 87a and 89a. Version 89a (July, 1989) allows for the possibility of an animated GIF, which is a short sequence of images within a single GIF file. A GIF89a can also be specified for interlaced presentation.. A patent-free replacement for the GIF, the PNG format, has been developed by an Internet committee and major browsers will soon be supporting it.

[8] IIS is a Web Server from Microsoft. It is a component of Microsoft's Windows NT 4.0 Server Operating System.

[9] A JPEG (pronounced JAY-peg) is a graphic image created by choosing from a range of compression qualities (actually, from one of a suite of compression algorithms). Along with the Graphic Interchange Format (GIF) file, the JPEG is a file type supported by the World Wide Web protocol, usually with the file suffix of ".jpg". One can create a progressive JPEG that is similar to an interlaced GIF.

MIME	Multipurpose Internet Mail Extensions[10]
NDI	Non-Developmental Item
NIA	Navy IETM Architecture
NLISP	Navy Logistics Information Strategic Plan
ODBC	Open Data Base Connection
PEDD	Portable Electronic Display Device
PDF	Portable Document Format[11]
PNG	Portable Network Graphics[12]
PURL	Persistent URL[13]
TCP/IP	Transmission Control Protocol/Internet Protocols[14]
URL	Uniform Resource Location.[15]
W3C	World Wide Web Consortium[16]

[10] MIME, a specification for formatting non-ASCII messages so that they can be sent over the Internet. Many e-mail clients now support MIME, which enables them to send and receive graphics, audio, and video files via the Internet mail system. There are many predefined MIME types, such as GIF graphics files and PostScript files. It is also possible to define one's own MIME types. In addition to e-mail applications, Web browsers also support various MIME types, enabling browsers to display or output files that are not in HTML format. See Internet URL: http://www.pcwebopedia.com/MIME.htm. The MIME-related Requests for Comments may be found at Internet URL: http://www.oac.uci.edu/indiv/ehood/MIME/MIME.html

[11] Adobe's Neutral Data Format for documents. Its Reader can be obtained from Adobe Systems at: http://www.adobe.com.

[12] PNG (pronounced "PING") is a file format for compressed graphic images that, in time, is expected to replace the GIF format that is widely used on today's Internet. The GIF format is patented by Compuserve and its usage in image-handling software involves licensing or other legal considerations.

[13] Short for Persistent URL, a type of URL that acts as an intermediary for a real URL of Web resource. When one enters a PURL in a browser, the browser sends the page request to a PURL server, which then returns the real URL of the page. PURLs are persistent because once a PURL is established, it never needs to change. The real address of the web page may change, but the PURL remains the same. See Internet URL: http://www.pcwebopedia.com/PURL.htm

[14] These are the basic communication protocols for the Internet. See: Postel, J., "Internet Protocol - DARPA Internet Program Protocol Specification", RFC 791, DARPA, September 1981; Postel, J., "Transmission Control Protocol - DARPA Internet Program Protocol Specification", RFC 793, DARPA, September 1981.

[15] A URL (Uniform Resource Locator) (pronounced "you-are-EL" or, in some quarters, "earl") is the address of a file or other resource accessible on the Internet. The type of file or resource depends on the Internet application protocol. Using the World Wide Web's protocol, the Hypertext Transfer Protocol (HTTP) , the file can be an HTML page (like the one you're reading), an image file, a program such as a CGI application or Java applet, or any other file supported by HTTP. The URL or resource address includes the name of the protocol required to access the file or resource, a domain name and, if it's a file, a hierarchical description of a file location on the server. Source: Internet URL: http://whatis.com/url.htm and RFC 1738 at Internet URL: http://andrew2.andrew.cmu.edu/rfc/rfc1738.html

[16] W3C Home Page. Internet URL: http://www.w3.org/

13. GLOSSARY

Active Server Pages is an open, compile-free application environment in which one can combine HTML, scripts, and reusable ActiveX server components to create dynamic and powerful Web-based business solutions. Active Server Pages enables server side scripting for IIS with native support for both VBScript and Jscript.[17]

Design-Time Web Controls Design-time Web controls are standard ActiveX & trade; controls that have a special interface, called IActiveDesigner, that lets them generate text that is saved into a file by the editor and processed at runtime. The structure and content of the text that is generated is entirely up to the control—any text that can be inserted into the file by a standard text editor can be generated by a design-time control. Design-time controls are based on COM, so that they are easy to build, share, and host. They also differ from standard ActiveX controls in that they contain no binary runtime component—they're never "alive" when a page is being viewed. Instead, a user sees only their HTML output. [18]

Frames A feature supported by most modern Web browsers than enables the Web author to divide the browser display area into two or more sections (frames). The contents of each frame are taken from a different Web page. Frames provide great flexibility in designing Web pages, but many designers avoid them because they are supported unevenly by current browsers.[19]

JavaBeans JavaBeans is the platform-neutral, component architecture for Java. JavaBeans allows developers to create reusable software components that can then be assembled together using visual application builder tools, such as Sybase's PowerJTM , Borland's JBuilderTM, IBM's Visual AgeTM for Java, SunSoft's Java WorkshopTM and Symantec's Visual Cafe, and many others.[20]

---

[17] Microsoft Site Builder Network Feature Stories. Internet URL: http://www.microsoft.com/sitebuilder/archive/features/aspover.htm

[18] March, 1997, Microsoft Interactive Developer Column: Preview by Jay Massena. Internet URL: http://www.microsoft.com/mind/0397/preview0397.htm

[19] Internet URL: http://www.pcwebopedia.com/frames.htm

[20] Java Beans : The Only Component Architecture for Java™. Internet URL: http://www.javasoft.com/beans/