E-Gov Enterprise Architecture Guidance (Common Reference Model)

Draft – Version 2.0

FEA Working Group
July 25, 2002

Endorsed by the Architecture and Infrastructure Committee

Federal CIO Council

The undersigned Co-Chairs do hereby endorse this E-Gov Enterprise Architecture Guidance (Common Reference Model) Draft 2.0 and consider it a roadmap for the Federal Government in achieving more effective and interoperable E-Gov technology solutions to meet business mission needs.

Signed: July 25, 2002

Lew Sanford, Jr.
GSA Chief Architect
General Services Administration
Co-Chair, Architecture Working Group

Michael Tiemann
Senior Technical Advisor,
Office of the CIO
Department of Energy
Co-Chair, Architecture Working Group

Lee Holcomb
Chief Information Officer
National Aeronautics and
Space Administration
Co-Chair, Architecture and
Infrastructure Committee
Federal CIO Council

John Gilligan
Chief Information Officer
United States Air Force
Co-Chair, Architecture and
Infrastructure Committee
Federal CIO Council

Credits

Thanks to the following individuals for participating in work sessions and contributing drafting, editing and feedback and assisting with delivery of a timely project.

Robert Benedict, NASA
James Benson, OMB-BAH
Carl Creager, DOT
Robert Haycock, OMB
Roopangi Kadakia, GSA
Barbara LaCour, USDA
Dawn Leaf, Smithsonian
Chris Meers, DOT
Roxie Murphy, GSA
Marion Royal, GSA
Numerous members of the Industry Advisory Council (IAC), including:
- Dan Twomey, Altarum
- John Dodd, CSC
- Andy Dziewulski, SAIC
- Mark Nelson, ICH
- Rick Smith, ICH
- Dan VanBellenghem, SRA
- V R Puvvada, Unisys (primary IAC liaison)

A special thanks to the Smithsonian Institution and its CIO staff for the use of the Smithsonian's Application/Web Content Management Server Technical Model and Technical Reference Model in this document.

Table of Contents

List of Figures

Summary

The President's E-Government Taskforce identified 24 Presidential Priority E-Gov Initiatives that are potentially transformational in nature and offer the opportunity to simplify and unify processes used by the Federal Government. These Initiatives will enable the Federal Government to better serve the American public, promote interactions across governmental organizations, and perform business activities while continuously improving internal efficiency and effectiveness. The OMB's Federal Enterprise Architecture Program Management Office has continuing stewardship responsibilities for the E-Gov Initiatives, as they become the first real instantiation of the Federal Enterprise Architecture.

The purpose of this document is to provide augmenting architectural guidance to the official direction from the Federal Enterprise Architecture Program Management Office. It is intended to provide a consistent, industry-aligned approach for defining and communicating about the components needed to cost and plan E-Gov programs – both the 24 Presidential Priority E-Gov Initiatives and other E-Gov Initiatives across the Federal Government.

This document describes a Federal-wide E-Gov target conceptual architecture. The architecture is based on the business requirements derived from the initiatives as well as system engineering design best practices. It provides a workable description of the components needed by E-Gov Initiatives and business activities to move rapidly into the web service-enabled business transaction environment.

Introduction

E-Gov Initiatives require a flexible, comprehensive architectural model that supports development of complete requirements when planning, designing, and building major systems. This is essential if the Federal Government is to 1) leverage information technology investments and avoid unnecessary duplication of infrastructure and major components, 2) link business processes through shared, yet sufficiently protected information systems, and 3) leverage disparate business processes, services and activities that are located outside Agency boundaries.

The OMB's Federal Enterprise Architecture Program Management Office (FEAPMO) has continuing stewardship responsibilities for the E-Gov Initiatives (www.feapmo.gov). This is part of the FEAPMO's larger role in defining a Federal Enterprise Architecture (FEA) - a function-driven framework for describing the business operations of the Federal Government independent of the Agencies that perform them.

Development of the FEA commenced on February 6, 2002. The purpose of this effort is to identify opportunities to simplify processes and unify work across the Agencies and within the lines of business of the Federal Government. The outcome of this effort will be a more citizen-centered, customer-focused government that maximizes technology investments to better achieve mission outcomes. The FEA is a business-based framework for cross-Agency, Federal Government-wide improvement. It provides OMB and the Federal Agencies with a new way of describing, analyzing, and improving the Federal Government and its ability to serve the citizen.

The purpose of this document is to augment FEAPMO guidance to E-Gov Initiative Teams and other web-based development efforts involving or affecting the Federal Government. It is a result and reflection of the ongoing interaction and cooperation between the Federal CIO Council and the FEAPMO.

This guidance provides a consistent, industry-aligned architecture for definition of and communication about the components commonly needed to deliver E-Gov solutions. This architecture will help avoid pitfalls such as:

- Duplicative efforts;
- Failure to consider infrastructure requirements; and
- Implementing technologies that are not sufficiently flexible or scaleable to meet Federal E-Gov requirements.

This approach also increases the potential for meaningful collaboration by clearly identifying opportunities where shared elements of E-Gov solutions might occur.

The common reference models contained herein are intended to extend from and support the high-level business architecture as defined by the FEA. Each view is intended to feature and describe the logical relationships of E-Gov capabilities, processing/access flows, technologies, and components. The intent is not to overly constrict the solutions, nor to proffer a solution that may be defined and implemented in only one manner. In fact, this document attempts to keep the potential solution sets broad and robust, capable of applying new and better technologies as they are conceived and delivered. That is why the explanations for these models stress that one or more components or parts may be logically combined or configured somewhat differently in actual solutions.

This document begins with conceptual guidance in the form of general architectural guidelines and a consistent vocabulary for E-Gov solutions. It then provides pragmatic examples of target architectures and standards for different components of effective E-Gov solutions.

Scope of this Document

This document augments architecture guidance information contained within the:

- Federal Enterprise Architecture Framework (FEAF), version 1.1 (September 1999);

- Architecture Alignment and Assessment Guide (October 2000);
- A Practical Guide to Federal Enterprise Architecture (February 2001); and
- Federal Enterprise Architecture (1st components released July 2002).

Its focus is on E-Gov architectural guidance at the Application and Technology levels of the FEAF and the FEA, although Business, Performance, and Data Architectures are briefly discussed to provide the overall context. Its focus is also on the To-Be or target architecture rather than describing the current environment or the roadmap for transition from the current to target environments.

This document also makes no attempt to define the physical architecture – where components physically exist. The information presented is conceptual in nature. That is, components could be outsourced to a service provider or to another Federal Agency. Components may reside on one server or many servers. Similarly, while the conceptual picture only indicates one instance of a component, that component could be deployed at several points.

Similarly, this document does not address the "process" side of developing E-Gov solutions. All E-Gov projects (like any development project) should identify and define a System Development Life Cycle approach and apply it beginning with the planning stages of each project.

Appendix C will be used for input provided by the Industry Advisory Council (IAC) on various architectural areas addressed in this guidance. This input will be included to show industry thinking in these areas, and for consideration in future versions of this guidance or other FEAPMO and Federal CIO Council initiatives. Inclusion in Appendix C does not imply endorsement of the IAC's opinions or recommendations by the Federal CIO Council FEA Working Group or the FEAPMO.

Finally, the guidance in this document is descriptive and suggestive rather than prescriptive and does not imply mandatory requirements for E-Gov Initiatives.

Alignment with the FEA and FEAF

The Federal Enterprise Architecture (FEA) is a function-driven framework for describing the business operations of the Federal Government independent of the Agencies that perform them. The Federal Enterprise Application Framework (FEAF), V1.1 provides various approaches, models, and definitions for communicating the overall organization and relationships of architecture components required for developing and maintaining the FEA.

This Guidance was developed in accordance with the basic principles and structure defined in the FEA and FEAF. It identifies a core set of E-Gov architectural concepts and pragmatic examples for E-Gov Initiatives across the Federal Government. In the terminology of the FEAF, the 24 Presidential Priority E-Gov Initiatives represent "slivers" (parts of Segments) of the overall Federal E-Gov Enterprise Architecture. This guidance will help those "slivers" and other E-Gov Initiatives coalesce into an effective Federal E-Gov Enterprise Architecture.

Principles

The FEAF defined, and the Federal CIO Council adopted, principles that govern and represent the criteria against which all potential investment and architectural decisions are weighed. The FEAF principles are summarized here in order to emphasize their applicability and importance to this E-Gov guidance:

1. Standards. *Establish Federal interoperability standards*. The Federal Government should adopt and use voluntary industry standards in which the interrelationships of components are fully defined by interface specifications available to the public and maintained by group consensus. The Federal Government should acquire and integrate preponderantly only those components conformant to these standards specifications. Non-proprietary system architectures and solutions are the goal; however, initially only partially and selectively compliant systems may be attainable. The key requirement is that records created on Agency information technology systems must be free of proprietary software dependencies. For E-Gov solutions the focus of interoperability is moving towards Internet and Web standards, XML, portals, new integration models such as Message Brokers and XML Web Services, and increasing use of hosting or Application Service Providers. All of these help isolate Agencies from traditional interoperability issues of the underlying hardware and software platforms. An Agency CIO performance goal to achieve this end might read, "Eliminate use of proprietary software dependencies." A certification requirement could be added to current and future contracts.

2. Investments: *Coordinate technology investments with the Federal business and architecture*. Investment decisions must be based on

business and architectural decisions that are aligned with the business needs of the Federal Government. Since the power of E-Gov solutions often comes from integration across existing functional and organizational boundaries, investments must also be looked at in this context. This includes both vertical Agency investments across traditionally separate functions, and cross Agency horizontal investments supporting a Line of Business or common E-Gov function.

3. Data Collection: *Minimize the data collection burden*. Data standardization, including a common vocabulary and data definition, will be difficult to achieve but is critical. A common organization eliminates redundancy and ensures data consistency. This is particularly important for E-Gov solutions that cross traditional organizational and functional boundaries which previously represented separate islands of data. E-Gov solutions also often involve direct data collection through automated access by new constituencies, e.g., a citizen entering information on a web form or a business's or State Government's systems automatically feeding data to an E-Gov application through the Internet. Thus, the principle of "enter once, use often" must be addressed in a wider context than in the past.

4. Security: *Secure Federal information against unauthorized access*. Appropriate security monitoring and planning, including an analysis of risks and contingencies and the implementation of appropriate contingency plans, must be completed to prevent unauthorized access to Federal information. Information security must be ensured and increased, commensurate with increased access to Federal information - and E-Gov solutions present an unparalleled increase in access to Federal Information by both traditional and new users. The requirements for information security cannot be achieved merely by establishing a "trusted network." In the case of highly sensitive information, each electronic record must be secured individually from alteration and inappropriate access. Indeed, if public information is mixed in an electronic file (e.g., a database) with information that is sensitive and should not be freely shared, it is necessary to secure those records at the level of subcomponents within each record.

5. Functionality: *Take advantage of standardization based on common functions and customers*. Federal Agencies should develop or design reusable components, or purchase architecture components, recognizing that these items are designed to obtain a particular functionality. Standardization on common functions and customers will help Federal Agencies implement change in a timely manner. For E-Gov solutions, this applies both to components that support common E-Gov functions across Agencies, and components that are needed to support multiple E-Gov functions.

6. Information Access: *Provide access to information*. The Federal Government should encourage a diversity of public and private access methods for Government public information. This should include multiple access points, and support for informational, transactional, and analytical access. Accessibility involves the ease with which users obtain information. Information access and display must be sufficiently adaptable to a wide range of users and access methods, including formats accessible to those with sensory disabilities. Effective information access is particularly critical for E-Gov solutions which often involve unprecedented levels of direct access by new user communities – both internal and external to the Federal Government.

7. Proven Technologies: *Select and implement proven market technologies*. Systems should be developed based on global data classes and process boundaries. Systems should be decoupled to allow maximum flexibility. Incorporating new or proven technology with full consideration to risk mitigation strategies will help Agencies to cope with change. This principle is particularly challenging in the E-Gov environment where technologies change and proliferate constantly. E-Gov solutions need to focus on emerging mainstream technologies with wide industry and government support.

8. Privacy: *Comply with the Privacy Act of 1974*. A privacy notice that includes the purpose for the information request should be provided anytime the public provides or enters data. The public should be given the right to choose whether or not to provide information. When information is used for other purposes than originally intended, an alternative privacy notice should be provided. Again, the public should be allowed to choose whether or not to provide the information. Protecting the privacy of the citizen is a tremendous burden and management must consider the potential uses of information. In addition, privacy information maintained by the Government should be properly secured. Again, privacy issues are particularly critical for E-Gov solutions which may combine previously separate data in a cross function/organization solution, and which may involve direct automated interaction with the public.

Architectures and Models

The FEAF defined, and the Federal CIO Council adopted, a four layer, segmented structure for defining the Federal Enterprise Architecture. Figure 1, shows the four layers of the FEAF and the corresponding models addressed in this guidance.

The models in this guidance associated with the Business, Performance (see Figure 2), Data, and Application Architectures are primarily conceptual

descriptions to establish a baseline of effective E-Gov architectural concepts and a common vocabulary. The models and standards associated with the Technology Architecture present more pragmatic guidance and examples for E-Gov Initiatives.

Business Architecture presents the evolving Federal Enterprise Architecture Business Reference Model that systematically identifies the business functions of the Federal Government. This model is provided for context and the guidance does not attempt to define business architectures or E-Gov processes for specific functions or organizations.

Data Architecture development was not practical in the timeframe available for this initial guidance. Instead, the Data Architecture section provides initial guidance on areas such as the use of the eXtensible Markup Language (XML), which are key to E-Gov solutions.

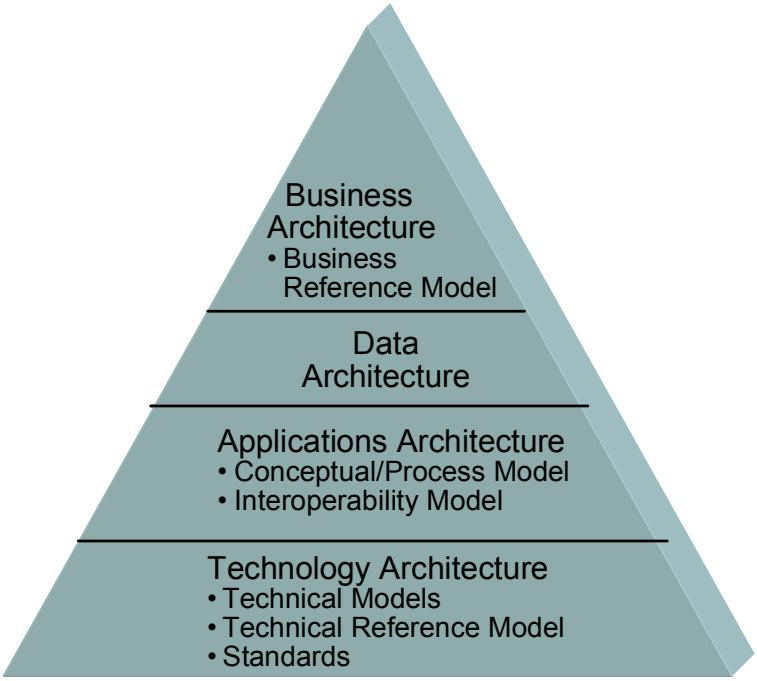

FIGURE 0 – ARCHITECTURES AND MODELS IN THIS GUIDANCE

Application Architecture defines the major application components common to E-Gov solutions, and includes two models:

- The Conceptual/ Process Model provides the bridge between the business view of the Business Reference Model and the systems view of the remaining models; and
- The Interoperability Model describes the common technical components of an E-Gov solution and how they interoperate within and across E-Gov solutions.

Technology Architecture provides more pragmatic implementation guidance for E-Gov Initiatives in the form of:

- Example Technical Models for major components of an E-Gov solution;
- E-Gov Technical Reference Model; and a
- "Starter set" of voluntary industry standards that should be understood and considered by E-Gov Initiatives.

Figure 2 shows how the FEA couples E-Gov architectures with the FEAF and government-wide reference models as the foundation for defining and implementing E-Gov cross-Agency solutions. The FEA includes a Performance Reference Model (PRM) that provides common outcome and output measures throughout the Federal Government. The PRM is further described later in this document.

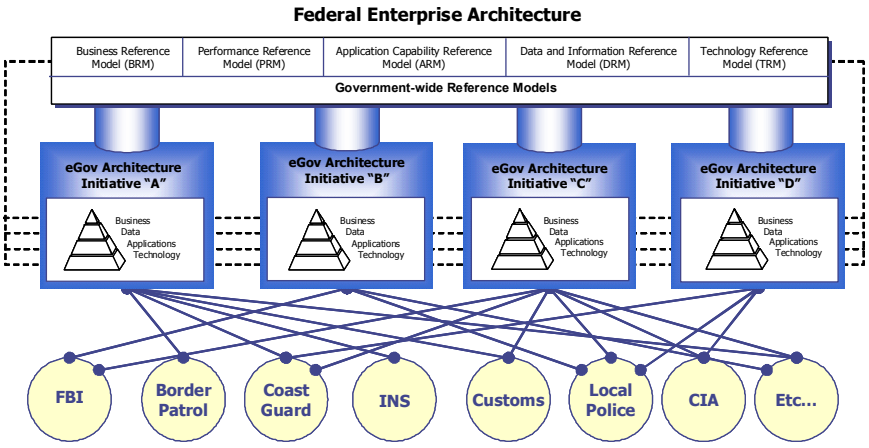

FIGURE 0 - FEDERAL ENTERPRISE ARCHITECTURE

Business Architecture

The Business Architecture identifies the functions, process, organization, and information flow for accomplishing the mission of an organization. E-Gov solutions often involve business solutions that cross traditional functional or organizational boundaries – both within and across Agencies, and with outside constituencies such as citizens, State and Local Government, and industry.

Figure 3 shows a very high level Business Architecture for the Federal Government. The bottom three tiers of the figure show the Federal Enterprise Architecture Business Reference Model (BRM). A description of the BRM and all of its components can be found at www.feapmo.gov.

The BRM provides an organized, hierarchical construct for describing the day-to-day business operations of the Federal Government. While many models exist for describing organizations – org charts, location maps, etcetera – this model presents the business using a functionally driven approach. The Lines of Business and Sub-functions that comprise the BRM represent a departure from previous models of the Federal Government that use antiquated, stove-piped, Agency-oriented frameworks.

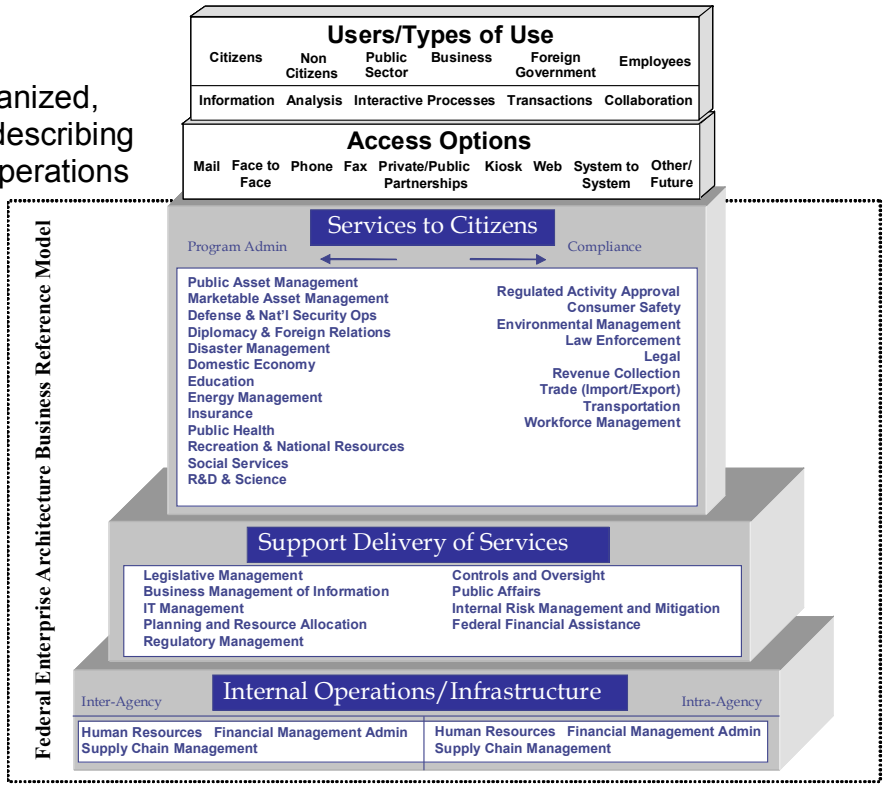

FIGURE 0 - BUSINESS REFERENCE MODEL

The BRM is the first layer of the Federal Enterprise Architecture. It is the main viewpoint for the analysis of data, applications and their capabilities, and the implementation of technologies to support reuse and standards. This framework should be used by Agencies when identifying and building E-Gov architectures to ensure that investments leverage existing components, applications, and services across the Federal Government.

The upper two tiers of Figure 3 address user access to the functions of the Federal Government defined in the BRM. A key consideration for E-Gov

Initiatives is the potential wide variety in: users, the kinds of interactions they need, and the kind of access methods they use.

The overall FEA addresses all of the potential access methods to ensure that the same comprehensive, consistent services are available no matter how access is achieved:

- Personal contact – such as face-to-face, voice (telephone, interactive voice response), videoconference;
- Electronic – such as facsimile, web browser, kiosk, system to system;
- Paper – such as traditional mail; and
- Service Providers – such as commercial vendors, private/public partnerships.

Because this Guidance is focused on E-Gov solutions, it addresses only web/internet based access approaches such as web browsers, e-system to e-system, and emerging devices such as Personal Digital Assistants and e-capable phones. However, in many cases, the "back-end" layers of the E-Gov architecture described in later sections should also provide a common architecture and infrastructure for supporting other access methods such as call centers.

E-Gov Initiatives should define their Business Architectures in terms of the BRM functional model and explicitly address the key issues of: types of users, types of use, and types of access methods to be supported.

Performance Measures Reference Model

The Performance Measures Reference Model (PRM) is a framework for performance measurement that provides common outcome and output measures throughout the Federal Government. It allows Agencies to better manage the business of Government at a Federal strategic level while providing a means for gauging progress towards the target FEA.

The PRM accomplishes these goals by establishing a common set of general performance outputs and measures that Agencies use to achieve much broader program and business goals and objectives. The model articulates the linkage between internal business components and the achievement of business and customer-centric outcomes. Most importantly, it facilitates resource allocation decisions based on comparative determinations of which programs/organizations are more efficient and effective.

The PRM will be designed to integrate with and complement OMB's development of the Program Assessment Rating Tool (PART) and common measures initiative. By defining outcome and output measures for lines of business and sub-functions, the PRM will provide the tools necessary to measure cross-agency initiatives at the Federal enterprise level.

Additional guidance on both the PRM and PART will be provided as these two models undergo continued development.

Data Architecture

The Data Architecture defines the major types of data needed to support the business, its meaning, and its form. Common data vocabulary and definitions are especially critical for E-Gov solutions which frequently cross traditional organizational (Federal and external), functional, and system boundaries. This includes not only operational data, but also analytic data, and web "content".

Each Line of Business and/or cross cutting E-Gov Initiative has a relevant set of data models and standards which need to be defined and applied to that function/initiative. This includes business driven common requirements for Privacy, Security, 508 access, and Records Management. E-Gov Initiatives should use the Unified Modeling Language (UML) for documenting these data models and common requirements. This will help foster consistency and integration across Initiatives.

Development of an E-Gov data architecture and related data models and standards was not attempted in the timeframe available for this initial guidance. However, the following subsections describe several Data Architecture considerations that are especially important for E-Gov Initiatives.

Data Interoperability Principles

E-Gov Initiatives should consider the following data principles aimed at increasing interoperability:

- Avoid non-standard data syntaxes;
- Seek industry vocabularies prior to the development of custom schemas – use these industry vocabularies as a starting point;
- Avoid creating a "one size fits all" schema – segment schemas into manageable efforts with business champions focused on expansion and government-wide propagation;
- Register the semantics of shared data elements; and
- Document service interfaces in a standard (consistent) way.

XML

The eXtensible Markup Language (XML) provides a critical foundation for E-Gov data architectures. XML is emerging as the Industry and Government standard for moving and sharing information – both among different entities and systems, and even among components of a system. XML provides an opportunity for Federal Lines of Business to define and standardize XML schemas for their functions and for interactions with other Lines of Business and external entities such as State and Local Governments or Industry. This will be particularly powerful where Lines of Business can leverage emerging industry standards such as ebXML, or join with State and Local Governments to define joint XML schemas that provide data interoperability across the tiers of government.

All E-Gov Initiatives should define and implement an approach for using XML. Where new development or re-development are pursued, XML should be considered for use as the default format for highly structured data as well as relatively less highly structured information, particularly at the User Interface layer but also at the Enterprise Repositories level as well. For legacy repositories that do not directly support XML, legacy to XML mapping and data transformation can be used to support interoperability across the data architecture. Use of voice XML (VXML) should be considered at the user interface level, especially for Government to Citizen (G2C) initiatives.

E-Gov Initiatives should work with communities in the relevant Lines of Business to define Federal-wide XML standards for their Line of Business. Where possible, these standards should leverage XML data elements and schemas that have been specified by voluntary consensus bodies as commercial and industrial standards.

E-Gov Initiatives and Lines of Business should register their XML schemas in a Federal-wide XML registry. This registry would support the development, registration and extension of XML schema, XML data element definition and naming conventions for "Inherently Governmental" data, and would facilitate public–private partnerships and collaboration in this critical area. Only the representations of the elements and schemas would be registered and available in the repository. The actual "instances" of data would be retained in the host system. This would promote standardization of data while leaving maintenance for the actual data with the appropriate Agency or Line of Business system.

A Federal-wide XML.gov Registry is currently being piloted (http://xmlregistry.nist.gov/xml-gov/), with implementation of an operational registry being considered for the FY2004 budget cycle. E-Gov Initiatives should begin the process of defining and registering their XML schemas with the pilot registry in preparation for Federal-wide rollout.

Privacy

E-Gov solutions' power to integrate data across traditional stovepipes and directly interact with the public also increases potential Privacy concerns. E-Gov Initiatives need to identify Privacy Sensitive data elements in advance of deployment to ensure that they are handled and safeguarded according to applicable regulations. The XML Registry should identify Privacy Sensitive data, and E-Gov solutions using Privacy Sensitive XML schemas should form communities of interest to develop and implement consistent, effective safeguards in accordance with the Privacy Act of 1974. Agencies that implement Privacy Sensitive data in their IT systems should be held accountable for effectively securing them from inappropriate use, while at the same time efficiently sharing them for purposes such as homeland security, and reducing

needless burdens upon the public to supply the same data to multiple stovepipe systems (in accordance with the Paperwork Reduction Act. of 1995.)

Physical Data Integration

E-Gov solutions often require data that come from multiple back end systems and databases associated with multiple organizations and functional areas. There is no single "right" answer to providing integrated data. The appropriate architecture will depend on the type, volume, and nature of the information being shared:

- Analytic data might be combined from multiple operational data stores into a hierarchy of data warehouses using a variety of data extraction and movement tools. Or, it could be supported through virtual databases which leave the data in its original location, but provide the appearance of an integrated database;
- High transaction volume interaction of operational data might be supported through a message broker infrastructure that links E-Gov and legacy applications. A distributed message broker infrastructure could combine message brokers into wider levels of integration; and
- An interactive E-Gov solution might require extracting parts of legacy operational databases and combining them into a new hybrid operational database with synchronization to the underlying operational systems.

As E-Gov Initiatives define their Data Architectures they should ensure that the Data Architecture can be deployed effectively and that appropriate data integration infrastructure is identified and implemented.

Application Architecture

The Application Architecture defines the applications and supporting capabilities to effectively manage the data and information needed to support business and performance objectives. This guidance focuses on the definition of a "building block" framework to support the reuse and assembly of application architecture components that leverage common services and functional capabilities.

The FEAPMO is in the process of defining an Application-Capability Reference Model (ARM). The ARM will be a business-driven, functional framework that classifies application capabilities with respect to how they support the business and/or performance objectives. The ARM will be structured across horizontal Service Areas that, independent of the business functions, can provide a leveragable foundation for reuse of applications, application capabilities, components, and business services. As the ARM matures, it will provide additional connectivity and traceability between the Business and Performance Reference Models and the Application Architecture.

Conceptual/Process Model

The Conceptual/Process Model provides the bridge between the purely functional view of the Business Reference Model and the more system/technology focused models that follow. Figure 4 shows the Conceptual/Process Model for E-Gov solutions. The architectures for E-Gov Initiatives should address all six layers of the Conceptual/ Process Model.

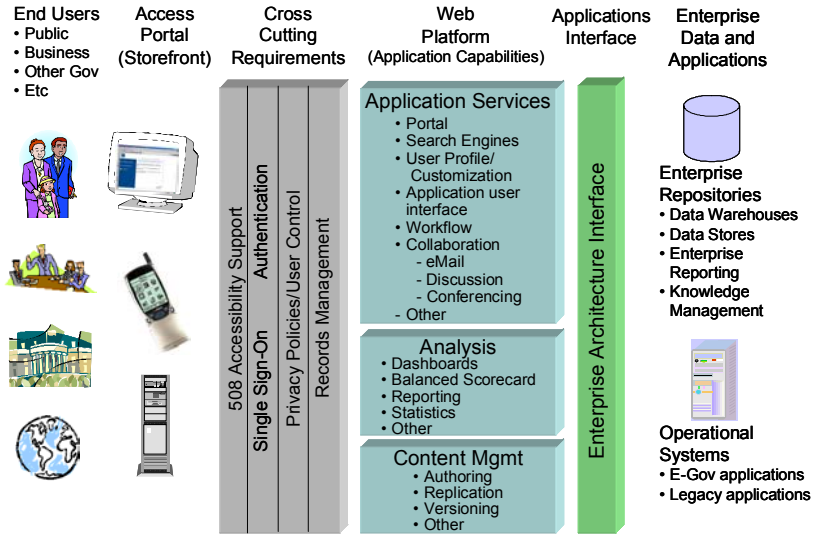

FIGURE 0 - CONCEPTUAL/PROCESS MODEL

End Users represent the variety of users described in the Business Reference Model (Figure 3). E-Gov Initiatives should understand their target user population and the ways in which they will use the E-Gov solution (e.g., information access versus financial transactions).

Access Portal represents the web/internet based access approaches such as web browsers, e-system to e-system, and emerging devices such as Personal Digital Assistants and e-capable phones. E-Gov Initiatives should determine which access methods are needed both now and in the reasonable future. They should also determine which non E-Gov access

methods (e.g., phone, regular mail) provide similar support for similar users and determine where E-Gov and non E-Gov solutions can leverage a common back end infrastructure.

The remaining four layers show a different view than the Business Reference Model. Instead of showing the Lines of Business of the Federal Government, they begin to lay out the application functions needed to support those Lines of Business.

Cross Cutting Requirements are common to all E-Gov application components in order to meet regulatory requirements and user expectations:

- Accessibility for all users in accordance with Section 508 of the Rehabilitation Act: Electronic and Information Technology Accessibility Standards;
- Single sign-on (access control) using secure authentication (per E-Authentication Initiative) to all needed E-Gov applications;
- Compliance with the Privacy Act of 1974 including disclosing E-Gov solutions use of privacy sensitive data and providing users appropriate control over the use of their data; and
- Maintaining records in unaltered form for as long as necessary to protect the rights of citizens as well as to provide access to the valuable information gathered and created using government systems.

Web Platform application components support interactions with users:

- Application Services components provide common web capabilities such as portal access with personalization for individual users and collaboration either synchronously through conferencing or asynchronously through email and discussion groups;
- Analysis components provide the capabilities for users to flexibly analyze and report on data – beyond the capabilities implemented in specific functional applications; and
- Content Management components provide E-Gov solutions with the ability to create, deploy, and control the broad range of textual and multimedia content typical of many E-Gov solutions.

Applications Interface application components provide a scalable mechanism for integrating the Web Platform with enterprise repositories and operational systems that serve business applications. The goals of the Applications Interface layer are to minimize the requirements for development of multiple custom point-to-point integration solutions and to minimize the impact to existing and future applications. The Applications Interface must be scalable to accommodate anticipated processing needs and include robust security functionality to prevent compromise of other systems it integrates with. A key for E-Gov solutions is the Applications Interface component's ability to reach outside of the Agency or even Federal Government and connect with other Federal, Industry, or State and Local Government systems.

Enterprise Data and Applications application components provide the core of the Federal Government's data and business logic, including:

- Operational and analytical data; and
- Major Legacy and E-Gov applications.

Accessing and unlocking the power of these Enterprise Data and Application components is critical for the success of many E-Gov solutions.

Interoperability Model

The Interoperability Model describes the primary application components supporting the Conceptual/Process Model and how they interoperate within and across E-Gov solutions. This includes interoperability at the user, data, and application levels. The Interoperability Model reflects commonly found industry representations, embracing industry standards and best practices.

Many components of the E-Gov Interoperability Model will be required for all E-Gov Initiatives. However, the business requirements of each E-Gov Initiative will determine which components are most critical or central to that initiative. E-Gov Initiatives should identify the critical components for their business requirements and ensure that those components are robustly supported in their architecture.

Figure 5 shows the E-Gov Interoperability Model. The major components of the Interoperability Model are:

Web Browser provides a standard user interface to E-Gov capabilities for most users. Initiatives that support Public access should strive for compatibility with a wide set of browser products and user machines by avoiding proprietary extensions and mobile code that may be blocked by users.

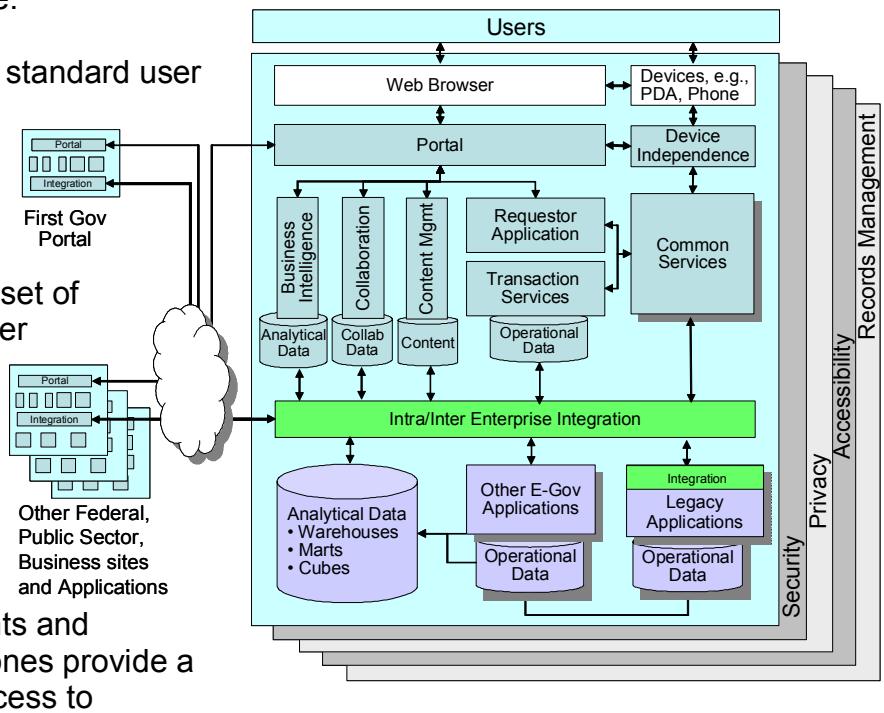

FIGURE 0 - E-GOV INTEROPERABILITY MODEL

Devices such as Personal Digital Assistants and web enabled cellular phones provide a growing capability for access to E-Gov capabilities from anywhere at anytime without

requiring a traditional Personnel Computer. There is a tremendous variety in available capabilities (processing power, screen sizes, input buttons) and connectivity options (different wireless network standards). Thus, it is critical that E-Gov Initiatives understand what kinds of devices need to be supported and how they will be used.

Device Independence components isolate E-Gov solutions from the complexity of the different devices and wireless networks and allow different devices to plug into the same underlying E-Gov capabilities.

Portals represent the leading concept for integrating many different information sources into a single mechanism for interacting with the user. They also facilitate providing services in a secured manner that can comply with Section 508 requirements. Multiple E-Gov Initiatives can share a portal, and multiple portals can be linked to integrate even more information sources and applications. The First Gov portal (http://www.firstgov.gov/) has been established as the root portal for the Federal Government.

Requestor Applications support the user interaction with E-Gov applications. This includes interaction with the portal, generating web pages for display on users' browsers, managing the user interaction, and accessing needed applications and data. The Requestor Application and/or the Transaction Services described below also maintain the user's "state" (e.g., using cookies, hidden fields, extended URLs) to overcome the underlying stateless nature of web interactions.

Transaction Services provide core business logic and performance, scalability, and reliability capabilities needed to support high volumes of transactions. They can be used to support reusable "business components" based on widely available industry component models.

Content Management provides capabilities to manage the large volume of "web" content (textual and multimedia) typical of many E-Gov Initiatives. This includes creation, storage and management of multiple versions of content along with branding and appearance templates to standardize the appearance of the content. One potential future trend is towards Enterprise Content Management that integrates content management, data management, and records management into one set of components.

Collaboration provides for both synchronous (e.g., video/audio conferencing, shared applications/whiteboard) and asynchronous (e.g., email, discussion groups) collaboration among users – both internal and external to the Government.

Business Intelligence provides capabilities to flexibly analyze and report on structured data beyond the capabilities built into specific functional applications.

This can range from ad hoc reporting, to Online Analytical Processing (OLAP) of multi-dimensional data, to sophisticated data mining or statistical analysis. For E-Gov Initiatives which cross traditional functional, organizational, or system stovepipes, this may required creation of an analytical data store combining data from multiple existing data warehouses or operational systems.

Common Services include directory, time, naming, and other services required for an interoperable distributed systems environment. They may also include specific services to support cross cutting requirements such as security (e.g., access controls, privacy rules, logging, and E-Authentication services).

Intra/Inter Enterprise Integration provides the backbone linking E-Gov solutions to other E-Gov solutions and legacy applications and data both within and outside the Federal Government. Many E-Gov Initiatives involve value chains that cross existing functional, organizational (including outside organizations such as State Government or Industry), and system boundaries. This may require integrating multiple systems (in multiple organizations) on a near real time basis, and with transactional robustness. The resulting E-Gov solution is really a composite application that combines processing and data capabilities of multiple systems into one end-to-end solution. Intra/Inter Enterprise Integration Components must support connectivity to commercial applications (such as Enterprise Resource Planning packages), web/e-Commerce vendor solutions, communications standards (such as SMTP and http), middleware, commercial data base packages, and a variety of older and sometimes antiquated Agency legacy applications. Since the late '90s, Message Brokers have been the dominant architecture for this kind of integration within enterprises, especially where high volumes of transactions were required. Message Broker architectures have also been extended to inter-enterprise integration using XML and web transport standards. XML Web Services are an emerging Intra/Inter Enterprise Integration approach to allow one application to discover and use the capabilities of another application. (In some cases these technologies are combined with Message Brokers using XML Web Services to connect with E-Gov applications.) Web based E-Gov solutions can also be integrated directly at the Portal or Requestor Application component level without going through a robust Intra/Inter Enterprise Integration component. Finally, virtual data base architectures can provide E-Gov Initiatives with access (typically read only) to multiple operational data bases if this level of data integration is all that is required.

Other E-Gov Applications will be involved in creating composite applications that combine capabilities and data from multiple E-Gov Initiatives. These other E-Gov Applications might be integrated through the Portal or Requestor Application components, as well as through the Intra/Inter Enterprise Integration component. Other E-Gov Applications might also share common components such as Portals, E-Authentication services, or Message Broker backbones. Operational data in Other E-Gov Applications may also be externalized and

aggregated/transformed in data warehouses (marts, cubes, etc.) to support sophisticated analyses.

Legacy Applications contain the vast majority of the Federal Government's detailed business logic and operational data. Integrating the business functionality and/or data from these systems will be the key to many E-Gov Initiatives. In some cases this can be accomplished through periodic extraction of data into data warehouses or combined operational data stores. However, in many cases, near real time application level integration will be required to support composite E-Gov solutions. Many legacy systems were not architected and implemented with this type of integration in mind. Thus, some sort of integration interface component may be required for these systems to interoperate with the Intra/Inter Enterprise Integration components.

Analytical Data components represent data warehouses (data marts, etc) of aggregated and transformed data from operational systems. Often there will be a hierarchy of data warehouses of progressively higher functional or organizational scope. Cross functional/organizational E-Gov Initiatives may be able to use existing data warehouses directly or need to aggregate their own layer of analytical data for the Business Intelligence component.

First Gov/Other Sites and Applications may interoperate with E-Gov applications to create solutions that span multiple applications and organizations. Interoperability can occur through links among multiple Portals (with First Gov being the portal of portals for the Federal Government), through the Requestor Application for web based applications, or, where robust transactional integration is required, through the Intra/Inter Enterprise Integration components.

Four components of the Interoperability Model reflect cross cutting requirements to meet regulatory requirements and user expectations.

Security architecture must be addressed from the beginning for every component of the Interoperability Model, from:

- E-Authentication Common Services; to
- Single sign on through the Portal; to
- Access control by Requestor Application and Transaction Services; to
- Encryption of network communication to the Browser; to
- Logging of Intra/Inter Enterprise Integration messages and Legacy System database updates; to
- Firewalls that protect the physical environment.

Security Management involves much more than just identifying and implementing the right technical application components. An appropriate security management plan, including items such as risks analysis, standard operating procedures, proper access controls, and business continuity, should be completed at all levels of the enterprise architecture (NIST 800-18 provides guidance on

preparing security plans). In order to properly manage security, a risk assessment and risk mitigation strategy should be developed, and the owner at each level of the enterprise architecture needs to understand and accept the residual risk. This process should be an integral part of the certification and accreditation of all E-Gov solutions.

Privacy similarly pervades the components of the Interoperability Model. For user interaction components it means explaining Privacy policies and what data will be used for and giving users the option to control use of their data. For application components it means controlling the access to Privacy Sensitive data and maintaining the integrity of that data. For integration and analytical components it means sensitivity to the privacy concerns of aggregating data from previously separate systems and organizations with potentially different privacy policies and safeguards.

Accessibility of E-Gov solutions is a requirement for all user facing components of the Interoperability Model (as well as supporting components such as documentation and training.) This means not only developing Section 508 compliant web pages, but selecting or developing products - such as Portal and Business Intelligence components, or functional products such as a Customer Relationship Management package - that support accessibility from the ground up.

Records Management ensures records (including email and increasingly multi media) are securely maintained in unaltered form for as long as necessary to protect the rights of citizens as well as to provide access to the valuable information gathered and created using government systems. The applicable records in all E-Gov data stores need to be identified, stored, scheduled, retrieved, transferred, destroyed, and securely controlled.

The boundaries of the different Interoperability Model components are not rigid. For example, some Intra/Inter Enterprise Integration capabilities may be present in the Transaction Services component. The components of the Interoperability Model reflect logical capabilities which may be implemented through a variety of products and technologies. Some existing COTS/GOTS or new development products may directly address a specific component (e.g., a Portal product). Others may combine parts or all of several components within a specific product. There is no one right mapping of Interoperability Model components to specific products. E-Gov Initiatives should consider several factors in defining specific products to support the components of the Interoperability Model:

- E-Gov Initiatives should identify the most critical/central components given their business requirements and consider selecting existing products or developing new products whose strength is focused on those components;
- Care should be taken in implementing existing products that cover multiple components just to use limited parts of their functionality (e.g.,

implementing an integrated Customer Relationship Management product for its automated email capabilities). While these products are becoming more modular, their key benefit is in their integration of a wide range of application components. The cost of implementing a robust product and integrating it into the architecture may outweigh the benefits of the limited functionality needed from the product; and

- Initiatives should look not just at how well a product supports a given component or set of components, but also how well it interoperates with all of the other Interoperability Model components and products required for the solution.

An even more fundamental concern is which components of the Interoperability Model should be addressed at the individual E-Gov Initiative level, and which should be addressed as common capabilities across E-Gov Initiatives. In most instances it is not preferable to have all of these components defined, architected, designed, and implemented on an Initiative by Initiative level. Instead many should be defined and/or implemented at higher levels in the chain. Some redundancy may prove necessary, but shared use of technologies is better for interoperability, usability and smart investment.

For example, the E-Authentication Initiative is building a Federal-wide authentication infrastructure that all E-Gov Initiatives should use to support authentication of users. Similarly, it would not make sense for each E-Gov Initiative to implement its own Portal component. Instead, most E-Gov Initiatives should "plug-in" to an existing Portal. This both reduces redundant costs and provides more of a "one-stop-shop" for users.

Figure 6 presents target levels for commonality of components across E-Gov Initiatives. It is based on three levels of commonality:

- Initiative Level – individual E-Gov Initiatives are responsible for architecting and implementing the component of the Inter-operability Model in accordance with the relevant Enterprise Architecture and this guidance;
- Common Component – a common product (or set of products) will be selected or developed for one or more components of the Interoperability Model. Individual E-Gov Initiatives are responsible for implementing

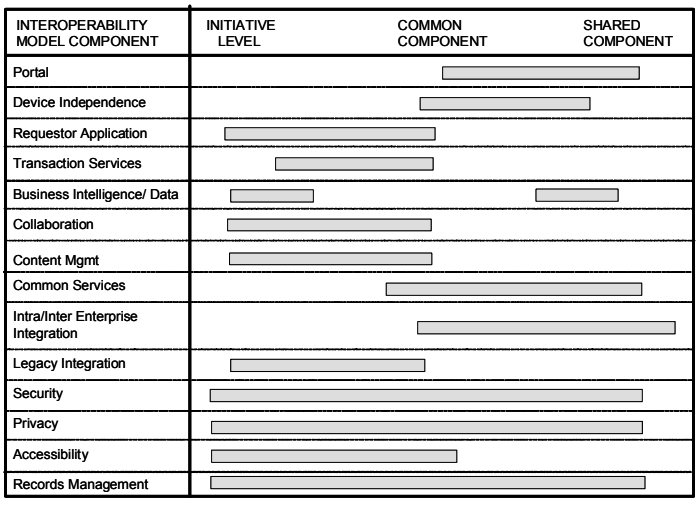

FIGURE 0 – INITIAL TARGET LEVELS FOR COMMONALITY OF INTEROPERABILITY MODEL COMPONENTS

that Common Component for their solution. At the smallest granularity this could be a reusable business logic component (e.g., for Grants Processing) which runs using the Transaction Services. At a higher level, it could be a specific portal product which all E-Gov Initiatives would implement in a standard way. At the highest level it could be a functional application (even an Enterprise Resource Planning package) that includes most components of the Interoperability Model; and

- Shared Component – parts or all of one or more components are provided as a shared infrastructure which multiple E-Gov Initiatives use. This is the model for E-Authentication and often for Portals. Similarly, the Integration component could be a Shared Component with multiple E-Gov Initiatives plugging into a central or distributed message broker infrastructure.

Identification and implementation of Common and Shared Components is being addressed by the Federal CIO Council and the FEAPMO. E-Gov Initiatives should architect their solutions so that they will be able to take advantage of Common and Shared Components as they become available.

Technology Architecture

The Technology Architecture defines the enabling hardware, software, and their physical locations to support the business applications/data and functions. The focus of this Guidance is on the E-Gov specific elements of the enabling software component of the Technology Architecture. It does not address the hardware view of the Technology Architecture, nor the physical architecture – where components physically exist. In other words components might exist at a Federal Agency or outsourced to a services provider; components may reside on one server or many servers; components may be centralized or distributed geographically. While not covered in this Guidance, E-Gov Initiatives should address these other dimensions of the Technology Architecture for their solution.

Technical Models

Technical Models provide examples of how different components of the Interoperability Model could be implemented with existing COTS/GOTS or new development. As described above, the Technical Models in this Guidance focus on E-Gov enabling software and do not address hardware or physical views.

Appendix A describes a variety of Technical Models addressing different parts of the Interoperability Model. Figure 7 presents an example of these Technical Models – application and content management components and services under a web platform. The major components of the example Technical Model are:

<u>Internet Application Server</u> supports the presentation and interaction with the user; transactional business logic; and supporting services that provide high security, performance, scalability, availability, and connectivity to existing operational data and systems. The dominant industry architectures for these types of these servers are:

- The J2EE (Java 2 Enterprise Edition) architecture from SUN which provides portability across multiple operating systems and hardware; and
- The .NET architecture from Microsoft which is currently focused on Microsoft operating systems running on Intel hardware.

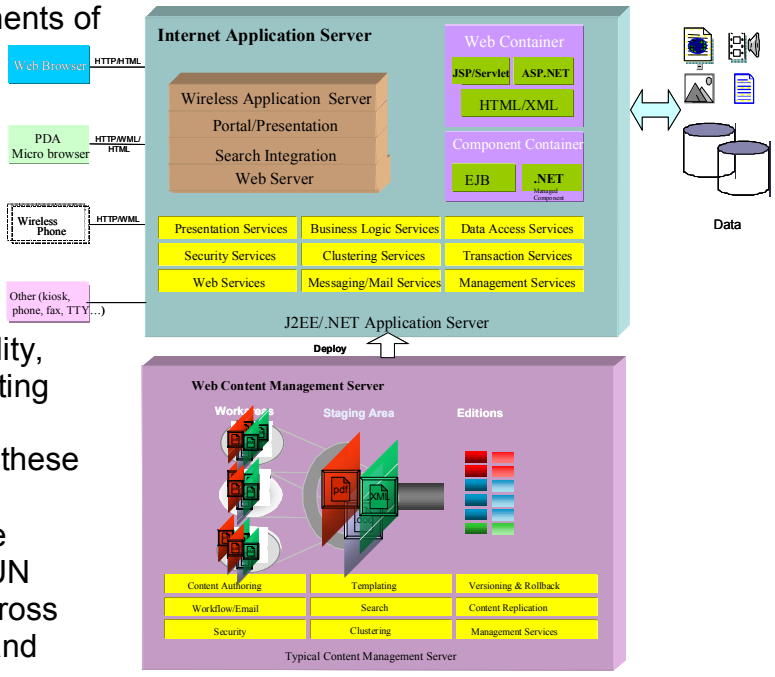

FIGURE 0 - SAMPLE TECHNOLOGY MODEL, WEB APPLICATION AND CONTENT MANAGEMENT

Leading Internet Application Server COTS products encompass many of the web/application layer components of the Interoperability Model. Alternatively, separate products are available for components such as the Portal or Web Server. As described in the discussion of the Interoperability Model, E-Gov Initiatives must carefully consider which components of the Interoperability Model they address in an integrated product and which they address by combining multiple more focused products. In general, integrated products or suites of tools may be preferred where several tools are needed to establish the functioning "web application," whereas less integrated or single purpose tools may be appropriate if the proposed solution only requires a few functions.

Web Content Management Server allows the site manager(s) to automate, manage, and control various aspects of content creation and formatting. They push this managed content out to an organization's Internet Application Servers, often in different formats supporting various browsing devices. At their core, web content management systems address the issues of: change management, dynamic content, workflow, design templates, repositories, replication and deployment, personalization, security management, scalability, and integration and development tools. Web Content Management solutions may be focused on content creation, content delivery, or business analytics. A complete solution almost always includes features from all three areas. There are a wide variety of integrated COTS Web Content Management Server products available. Alternatively, basic Web Content Management Server capabilities may be bundled into another enabling product such as an Internet Application Server, or a business product such as a Customer Relationship Management product.

E-Gov Technical Reference Model

The Technical Reference Model (TRM) and associated Standards are integral components of enterprise architecture and are required by OMB A-130. The purpose of the TRM is to provide a conceptual framework or context in which to define a common technical vocabulary, so that Federal Agencies can better coordinate acquisition, development, and support for E-Gov Initiatives.

A TRM is a widely accepted "best practice" and there may be different TRMs depending on the scope and complexity of the concerned Agency. The National Institute of Standards and Technology (NIST) OSE Reference Model in conjunction with the Open System Interconnection (OSI) Service Layer Model provide the initial foundation for many TRMs across the Federal Government, e.g., Department of Energy, Department of Labor, Department of Education, Smithsonian, Department of Defense.

Figure 8 shows the E-Gov TRM built on the OSE/OSI foundation. It provides a "working" model or framework that is neither overly prescriptive nor overly

general for a cross Agency, cross E-Gov Initiative model. The E-Gov TRM includes the following elements:

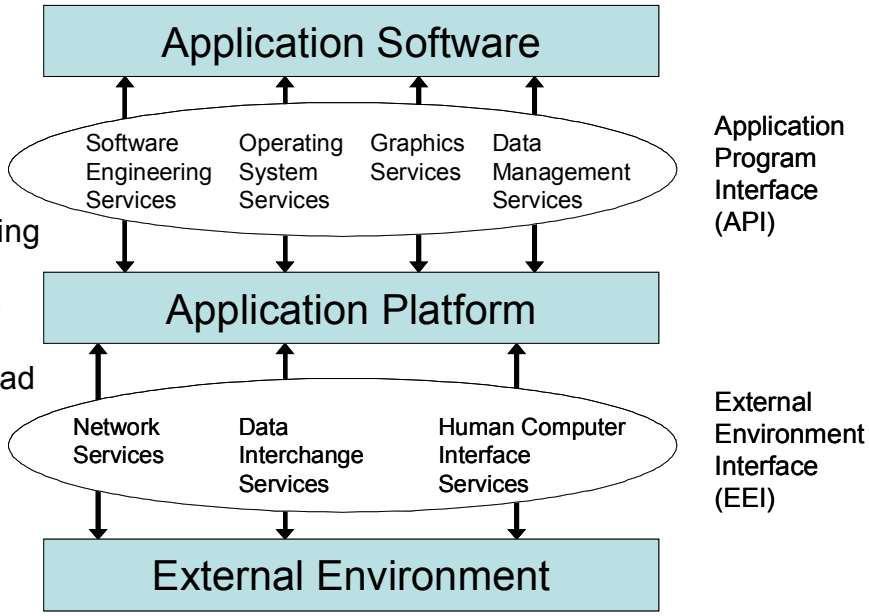

FIGURE 0 - E-GOV TECHNICAL REFERENCE MODEL

Application Software includes data, documentation, and training as well as programs. A key goal is development and reuse of modular application components to support the broad range of E-Gov activities that are common within and across Agencies and Lines of Business. Previously developed reusable code components, and GOTS or COTS products should be identified and registered for consideration in other E-Gov Initiatives. Such reusable components would then be integrated with any other pieces needed to satisfy all of the requirements for those E-Gov Initiatives.

Application Platform is composed of the collection of hardware and software components that provide services or software resources for the application software. As much as possible, the implementation-specific characteristics of the Application Platform should be transparent to the application software.

External Environment consists of those system elements that are external to the application software and the application platform (e.g., services provided by other platforms or peripheral devices). These entities are classified into general categories of: users, information interchange entities, and communications entities.

Application Program Interface (API) is the interface between the application software and the application platform. Its primary function is to support portability of application software. An API is categorized in accordance with the types of service accessible via that API: human/computer interface services, information interchange services, communication services, and internal system services. The emergence of J2EE and .NET as dominant architectures, along with J2EE's cross operating system/hardware portability, and increasing use of hosting or Application Service Providers provides increasing levels of application portability, system interoperability, and scalability.

External Environment Interface (EEI) is the interface that supports information transfer between the application platform and the external environment, and

among applications executing on the same platform. Consisting chiefly of protocols and supporting data formats, the EEI supports interoperability to a large extent. An EEI is categorized in accordance with the type of information transfer services provided, to and from: human users, external data stores, and other application platforms. HTML/HTTP, XML, message brokers, XML Web Services, and other Internet based architectures are emerging as fundamental building blocks supporting EEI interchange and interoperability.

Services within the TRM include:

- Software engineering services – the infrastructure to develop and maintain software that exhibits the required characteristics;
- Operating system services – the core services needed to operate and administer the application platform and provide an interface between application software and the platform;
- Graphics services – the functions required for creating and manipulating displayed images;
- Data management services – the management of data that can be defined independent of the processes that create or use it, maintained indefinitely, and shared among many processes;
- Network services – the capabilities and mechanisms to support distributed applications requiring data access and applications interoperability in heterogeneous networked environments;
- Data interchange services – the specialized support for the exchange of information, including format and semantics of data entities, between applications on the same or different (heterogeneous) platforms;
- User interface services – the methods by which people may interact with an application; and
- Security and System Management services – these services are common to all of the service areas and pervade these areas in one or more forms.

Agencies are expected to develop their own TRMs which reflect their specific business requirements. For example, Figure 9 shows the Smithsonian's TRM with its explicit inclusion of Output Services, Information Exchange Media, and Information Storage Media to support its core museum and library business functions.

Further expansion or specification of the E-Gov TRM will take place in the future through CIO Council and FEAPMO initiatives to ensure effective reuse and interoperability across E-Gov Initiatives.

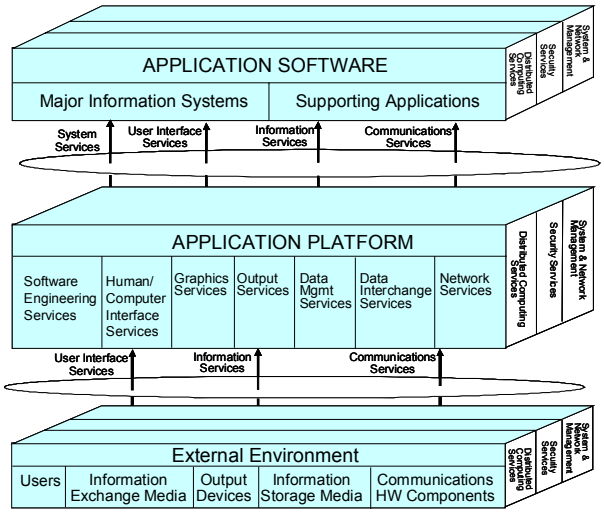

FIGURE 0 - TRM REFLECTING SMITHSONIAN'S SPECIFIC BUSINESS REQUIREMENTS

Standards

The world of E-Gov standards is rapidly changing and evolving. This is true both of the specific standards themselves, and of the nature of the standards - in particular “open” standards. "Open" means both available to all (at a reasonable fee) and a consensus process that is open to the entire industry. “Open” standards help ensure portability of applications and data, and help avoid over dependence on specific vendors. Most “open” standards have evolved through the wide participation of academia, business, government, and industry in various formal standards bodies. These include the International Telecommunication Union (ITU), the American National Standards Institute (ANSI), the National Institute of Standards and Technology (NIST), the INTERNET Engineering Task Force (IETF), and the Institute of Electrical and Electronic Engineers (IEEE). Standards developed through such organizations have a very wide base of support that significantly improves their viability in the marketplace.

However, many consortia involved in E-Gov standards are only “kind of” open. For example, the World Wide Web Consortium (W3C) uses an open process, but Tim Berners-Lee retains final decision authority. Some consortia are dominated by a few vendors, such as the Web Services Interoperability Organization where SUN was not a founding member even though J2EE has emerged as a major platform for XML Web Services. In some cases, vendors want to standardize the specification of a branded commercial product they offer without losing control of the product’s direction. For example, Sun uses many open processes, but retains final authority on Java.

This is the real world facing E-Gov Initiatives – a world built on a wide range of proprietary, quasi-open and fully open specifications. Thus, instead of “open standards”, this Guidance focuses on “voluntary industry standards” (the term used in the National Technology Transfer and Advancement Act) to meet the portability and interoperability goals of E-Gov Initiatives.

Appendix B presents an initial selection of E-Gov “voluntary industry standards” that should be understood and considered by Federal E-Gov Initiatives. The list in Appendix B is neither comprehensive nor prescriptive and has been only narrowly reviewed. It is designed solely to help inform and guide E-Gov Initiative architects. To make it easier to relate the items in Appendix B to the models in this Guidance, the standards are organized by Interoperability Model component, with cross reference to the E-Gov TRM.

More comprehensive and prescriptive standards guidance for E-Gov Initiatives will be addressed through ongoing CIO Council and FEAPMO initiatives.

Conclusion

The Presidential Priority E-Gov Initiatives and others within various Agencies across the Federal Government are simplifying and unifying the reach and range of services and support in interactions with business, other government enterprises and the public. The intent is to put in place solutions that are interoperable and reusable. This will improve services while reducing needless duplication and redundancy. To ultimately achieve this on a large scale will require continuous focus on a Federal Enterprise Architecture which affords the kinds of cross cutting capabilities and infrastructures required for these and future E-Gov initiatives to succeed.

The guidance in this document augments the guidance provided by the FEAPMO. It provides an initial set of terminology, architectural concepts, standards and technology models that provide a common foundation for E-Gov initiatives, both Government-wide and within various Agencies. This should facilitate the supportability of and interoperability among E-Gov Initiatives and capabilities.

This guidance is just a start, and will need to evolve and be updated. The continued cooperation and support of the Federal CIO Council, the FEAPMO, Federal Agencies, and industry (through the Industry Advisory Council and the various not-for-profit consortia) is essential. Working together, it is assured that a Federal Enterprise Architecture with significant E-Gov enhanced and more efficient and cost effective business capabilities will grow and succeed in unifying and simplifying Government.

Appendix A
Example Technical Models

The Technical Models in this appendix show examples of how different components of the Interoperability Model could be implemented. Additional examples will be added over time. As described above, the Technical Models in this Guidance focus on E-Gov enabling software and do not address hardware or physical views.

A-1 Example Message Broker Technical Model

A-2 Example XML Web Services Technical Model

A-1 Example Message Broker Technical Model

Since the late 1990s Message Brokers have emerged as the leading architecture for Enterprise Application Integration. The first role of Message Brokers was to tame the spaghetti tangle of traditional point to point interfaces (Figure 10). Instead of every system being directly interfaced with every other system, each system was simply interfaced to the Message Broker (Figure 11). This not only drastically reduced the number of needed interfaces, but also isolated each system from changes in the others. Instead of every interfacing system being affected when a system changed, only the one interface to the Message Broker was affected.

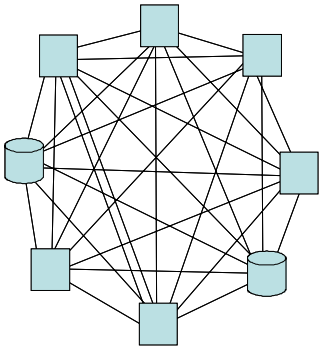

FIGURE 0 - INTEGRATION SPAGHETTI

The second role was to combine previously stovepipe applications and data into integrated applications to meet the cross function/cross organization requirements of Customer Relationship Management and web/E-Commerce. Instead of just providing data integration across systems, Message Brokers could use messaging to provide near-real-time transactional integration. This meant the functional capabilities of different systems could be combined into one composite application. Message Brokers could even reach outside the enterprise using Internet transport standards and connect to systems within other enterprises. To the user, the composite application appeared to be one integrated cross function/cross organization system (e.g., an end-to-end acquisition process). It was only from the inside that you could see the Message Broker orchestrating the different heterogeneous parts into the composite whole.

Today, Message Brokers can be interconnected to form larger and larger interoperable environments. The end result is what the Gartner Groups refers to as an Enterprise Nervous System – a Message Broker infrastructure that applications can plug into to provide a near-real-time integrated environment.

Figure 11 shows an example Message Broker Technical Model. It is composed of the following major components:

Message Broker shown in the middle integrates the heterogeneous applications and data stores shown around the circumference. The Message Broker provides three key services:

- Messaging, Data Movement provides physical transport of the messages and data among the applications. This can be done using Internet protocols such as HTTP, traditional messaging systems such as IBM's Message Queuing, sockets, or other communications mechanisms. The messages themselves were originally coded in different vendors' proprietary formats. Increasingly, however, Message Brokers support XML as their message language;

- Intelligent Routing determines which messages should go to which applications. This often includes Publish and Subscribe style routing where server applications publish a type of business event once to the Message Broker and multiple client applications that are interested in that type of event can subscribe to it;
- Transformation provides data mapping among the potentially different data syntax and even semantics of the different applications. Thus, if one application uses M/F for male/female and another expect 1/0, the Transformation layer of the Message Broker can map from one to the other without impacting the communicating applications. More complexly, if one application expects a customer entity to include five attributes, but another application divides those attributes across two entities, the Transformation Layer could map between these different structures.

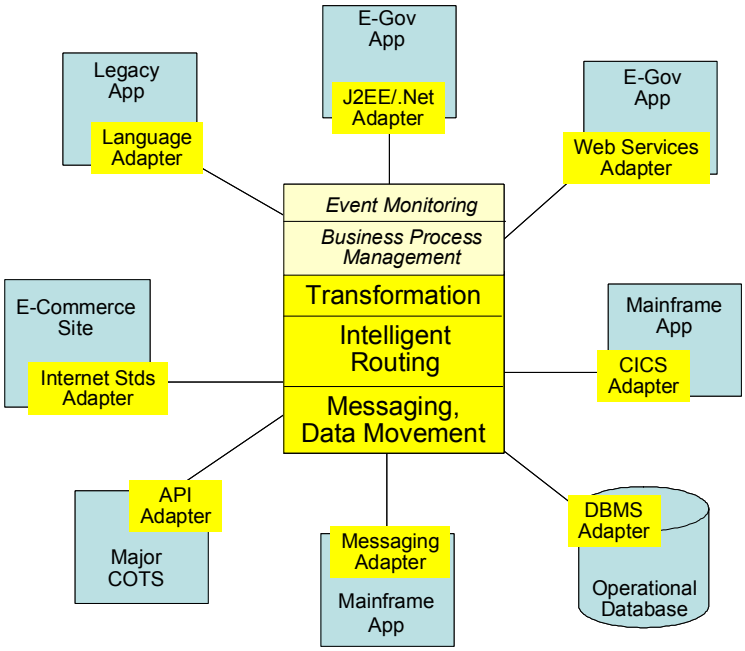

FIGURE 0 - MESSAGE BROKER TECHNICAL MODEL

The Message Broker architecture may also include two additional higher level services:

- Business Process Management takes Intelligent Routing to the level of complex cross application workflow that supports end-to-end internal and external processes; and
- Event Monitoring leverages the Message Broker's role as the center of information flow across the enterprise to support near-real-time analysis of business operations.

Adapters allow each of the applications/data stores to interact with the Message Broker. Different types of adapters allow Message Brokers to integrate a huge variety of heterogeneous applications:

- J2EE or .Net adapters can connect to E-Gov applications built on those distributed computing platforms;
- Web Services adapters allow message brokers to interact with XML Web Services;
- Messaging adapters can connect to existing messaging infrastructures, e.g., IBM's Message Queuing which is widely used in IBM mainframe environments;

- CICS or other mainframe transaction monitor adapters can provide direct connection to mainframe applications;
- Relational Database adapters can provide direct integration with databases without going through an application;
- API adapters are available out-of-the-box for many leading COTS products such as Enterprise Resource Planning or Customer Relationship Management packages. These adapters may directly use the APIs provided by the vendor or may even provide a higher level, easier to use API which then calls the vendor APIs;
- Internet Standards based adapters (increasingly being replaced by Web Services) can be used to integrate with E-Commerce sites outside the enterprise; and
- Language adapters, e.g., for Java, C, etcetera, can connect to almost any mid tier environment.

A-2 Example XML Web Services Technical Model

XML Web Services are an emerging architecture for allowing applications to discover and access functional capabilities of other applications using Internet based standards. (This guidance uses the term XML Web Services to distinguish this architecture from typical services provided by a Web platform, e.g., search capabilities.) XML Web Services were originally proposed as part of Microsoft's .Net architecture. Since then they have been adopted more widely, in particular with support from Sun's J2EE platform.

XML Web Services were designed from the beginning for the loosely coupled, inter-enterprise world of the Internet. Thus, they focus on 1) using Internet standard HTTP for transport through fire walls, 2) using XML as the standard data format, and 3) providing standard mechanisms for describing and finding available XML Web Services.

XML Web Services are most often used to link web applications such as E-Gov solutions, portals, or external web sites or e-systems. However, they can also be used to link to legacy applications and Message Brokers – tying together the web world and the message broker's integration of back-end applications. Unlike Message Brokers, XML Web Services generally provide direct synchronous connections between the client and server application, rather than asynchronous connection through a central hub.

Figure 12 shows an example XML Web Services Technical Model. Its main components include:

Applications include E-Gov Applications, Legacy Systems (generally through some sort of wrapper), Message Brokers, or other Government or E-Commerce sites. Applications may be 1) clients of XML Web Services provided by another application, 2) providers of XML Web Services to other applications, or 3) both. By using XML Web Services from another application, E-Gov Initiatives can deliver composite applications that combine the capabilities of multiple applications into one integrated E-Gov solution. This also reduces cost by reusing capabilities that are already provided in another application.

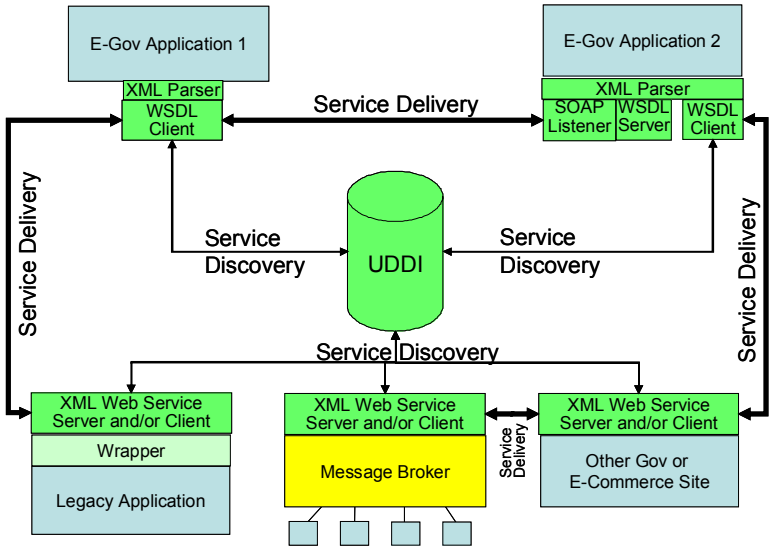

FIGURE 0 - XML WEB SERVICES TECHNICAL MODEL

Universal Description Discovery and Integration (UDDI) provides a registry of available XML Web Services. Applications can search the registry for desired services and obtain the URL of the service and a WSDL file describing the service and how to interact with it. Applications can also register the XML Web Services they provide in the UDDI registry.

Web Service Description Language (WSDL) provides an XML based description of XML Web Services and how to interact with them – it is the Interface Definition Language of XML Web Services. WSDL Clients obtain the WSDL file for an XML Web Service from UDDI, parse the XML with an XML parser, and use the information and the URL to call the Web Service using SOAP. WSDL Servers take SOAP messages from the SOAP Listener, parse the XML, and pass the Web Service request to the server application.

Simple Object Access Protocol (SOAP) is a lightweight remote procedure call protocol that uses XML for message formats and HTTP for transport. A SOAP method is simply an HTTP request and response that complies with the SOAP XML encoding rules. A SOAP endpoint is simply an HTTP-based URL that identifies a target for method invocation. While SOAP is typically implemented as a synchronous protocol based on HTTP, it can be implemented over other transports such as sockets or messaging systems if more robust or asynchronous communications are required.

Figure 13 shows how a combination of XML Web Services and Message Brokers could be used to create an integrated E-Gov solution.

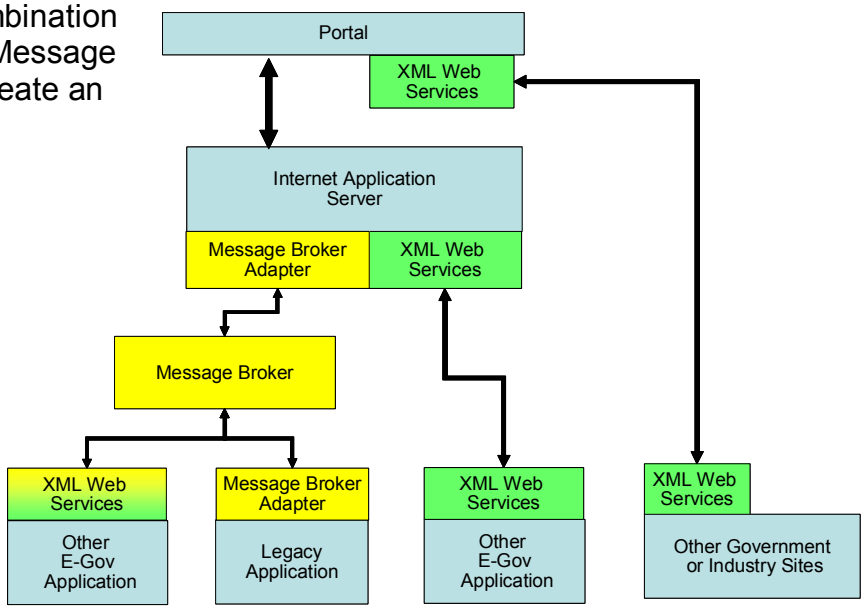

FIGURE 0 - COMBINING XML WEB SERVICES AND MESSAGE BROKERS FOR AN INTEGRATED E-GOV SOLUTION

Appendix B

Initial E-Gov Voluntary Industry Standards

This Appendix presents an initial selection of E-Gov "voluntary industry standards" that should be understood and considered by Federal E-Gov Initiatives. It is not the intent of this guidance to identify all possible standards applicable to E-Gov, nor to create even a comprehensive set. The list is intended as a starter set based on the judgment of the working group regarding their relevancy to E-Gov. It was intended that this serve as a point to begin considering relevant standards and to serve as a place wherein appropriate and obvious voluntary industry standards can at least be included in an architectural context. A quick scan of these standards will validate their applicability to E-Gov. Further initiatives under the CIO Council and Federal Enterprise Architecture Program Management Office will continue identifying both criteria for standards applicability and use, as well as the appropriate standards themselves.

To make it easier for readers to relate the standards in this Appendix to the models in this Guidance, the standards are organized by Interoperability Model component, with cross reference to the E-Gov TRM. Not all Interoperability Components have E-Gov voluntary industry standards included in this starter set.

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Browser	Human Computer Interface	A wide range of browser platforms should be supported, particularly for E-Gov solutions accessed by the Public. These include: • Netscape Navigator • Microsoft Internet Explorer
		HTML – Hyper Text Markup Language The language used to create Web documents and a subset of Standard Generalized Markup Language (SGML http://www.w3.org/MarkUp/
		DHTML - Dynamic HTML A collective term for a combination of new Hypertext Markup Language (HTML) tags and options, style sheets, and programming that will allow Web pages that are more animated and more responsive to user interaction than previous versions of HTML.
		XHTML – eXtensible HTML (emerging) The W3C's recommendation for the next generation of HTML leveraging XML http://www.w3.org/TR/2001/REC-xhtml11-20010531/

Model Component	TRM Service	Starter Set Voluntary Industry Standards
		There is a tremendous range of devices with different capabilities (processing power, screen size/resolution, input capabilities, and memory) and operating systems that E-Gov solutions may need to support.
		Palm Operating System Palm is the leading Personal Digital Assistant (PDA). Version 5 of Palm OS provides multitasking and other capabilities that will provide an improved platform for E-Gov solutions. http://www.palmos.com/dev/
		Pocket PC 2002 Microsoft's environment for PDA level devices. http://www.microsoft.com/mobile/pocketpc/learnmore.asp

		Pocket PC Phone Edition Microsoft's environment for internet capable cellular phones. http://www.microsoft.com/mobile/pocketpc/phoneedition/default.asp
		Blackberry The leading email enabled wireless device with wide use in several Agencies. http://www.blackberry.com/developers/na/index.shtml
		Symbian Epoc A leading environment for web capable cellular phones http://www.symbian.com/developer/index.html
	External/ Software Engineering	J2ME - Java 2 Platform, Micro Edition Sun's Java environment for devices. It promises a relatively portable environment for those using Java for other tiers of the architecture. http://java.sun.com/j2me/docs/

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Device Independence	Data Interchange Services/ Human Computer Interface Services	Because of the wide variety in devices, there are a variety of existing and emerging standards to provide portability of applications across different device platforms.
		WAP – Wireless Application Protocol The Wireless Application Protocol (WAP) is an open, global specification that empowers users of digital mobile phones, pagers, personal digital assistants and other wireless devices to securely access and interact with Internet/intranet/extranet content, applications, and services. http://www.wapforum.org/
		XHTMLMP – XHTML Mobile Profile (emerging) XHTMLMP is designed for resource-constrained Web clients that do not support the full set of XHTML features, such as mobile phones, PDAs, pagers and set-top boxes. It extends XHTML Basic with modules, elements and attributes to provide a richer authoring language. XHTML replaces the Wireless Markup Language (WML). http://www.wapforum.org/what/technical.htm

		VXML – Voice XML (emerging) VXML is an XML vocabulary for specifying IVR (Integrated Voice Response) Systems. http://www.w3c.org/Voice/ http://www.voicexml.org/
		CC/PP – Composite Capability/Preference Profiles (emerging) CC/PP framework specifies how client devices express their capabilities and preferences (the user agent profile) to the server that originates content (the origin server). The origin server uses the "user agent profile" to produce and deliver content appropriate to the client device. In addition to computer-based client devices, particular attention is being paid to other kinds of devices such as mobile phones. http://www.w3c.org/2001/di/

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Portal	Data Interchange Services	JSR 53 - Java™ Servlet Java™ Servlets provide reusable web components that can be incorporated into portals. http://www.jcp.org/aboutJava/communityprocess/final/jsr053/
		JSR 168 – Java™ Portlet API Java™ Portlet API enables interoperability between Portlets and Portals by defining APIs that address the areas of aggregation, personalization, presentation and security. http://www.jcp.org/jsr/detail/168.jsp
		WSRP – Web Services for Remote Portals (emerging) WSRP defines an XML and Web services standard that will allow the plug-n-play of visual, user-facing Web services with portals or other intermediary Web applications. http://www.oasis-open.org/committees/wsrp/

		WSUI – Web Services User Interface (emerging) WSUI uses a simple schema for describing a WSUI "component" that can be used in a portal to call backend SOAP and XML services. WSUI uses XSLT stylesheets to construct user-facing views to enable users to interact with the services. http://www.wsui.org/

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Requestor Application	Data Interchange Services	JSP – Java™ Server Pages JSP is part of Sun's J2EE architecture and provide template capabilities for presenting dynamically generated Web content. JSPs are text files written in a combination of standard HTML tags, JSP tags, and Java code. http://java.sun.com/products/jsp/
		ASP.Net – Active Server Pages.Net ASP.NET is a set of technologies in the Microsoft .NET Framework for building Web applications and XML Web Services. ASP.NET pages execute on the server and generate markup such as HTML, WML or XML that is sent to a desktop or mobile browser. http://msdn.microsoft.com/library/default.asp?url=/nhp/Default.asp?contentid=28000440

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Transaction Services	Data Interchange Services	J2EE – Java™ 2 Platform Enterprise Edition Sun's J2EE and Microsoft's .Net are the two dominant distributed computing architecture frameworks. J2EE provides portability of a single language (Java™) over multiple operating systems and hardware platforms. http://java.sun.com/j2ee/download.html#platformspec
		.Net Microsoft's .Net and Sun's J2EE are the two dominant distributed computing architecture frameworks. .Net supports a wide range of languages but is primarily tied to the Microsoft Windows operating system and Intel hardware. http://www.microsoft.com/net/products/default.asp

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Content Management	Data Management	WebDAV (RFC 2518) – World Wide Web Distributed Authoring and Versioning (emerging) WebDAV is an interface standard that defines the syntax used by an authoring tool when interacting with a Web server. It is a set of extensions to the HTTP protocol which allows users to collaboratively edit and manage files on remote web servers. http://www.webdav.org/ http://www.ietf.org/ids.by.wg/webdav.html

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Collaboration	Network Services	IMAP (RFC2060) V4.1 – Internet Message Access Protocol IMAP4rev1 allows a client to access and manipulate electronic mail messages on a server. IMAP4rev1 permits manipulation of remote message folders, called "mailboxes", in a way that is functionally equivalent to local mailboxes. IMAP4rev1 also provides the capability for an offline client to resynchronize with the server.
		MIME (RFC 2045) – Multipurpose Internet Mail Extensions MIME extends the format of Internet mail to allow non-US- American Standard Code for Information Interchange (ASCII) textual messages, non-textual messages, multi-part message bodies, and non-US-ASCII information in message headers. MIME support allows compliant email clients and servers to accurately communicate embedded information to internal and external users.
		SMTP (RFC821) – Simple Mail Transfer Protocol SMTP facilitates transfer of electronic-mail messages. It specifies how two systems are to interact, and the messages format used to control the transfer of electronic mail.
		ESMTP (RFC1869) – Extended Simple Mail Transfer Protocol ESMTP allows new service extensions to SMTP to be defined and registered with Internet Assigned Numbers Authority (IANA)

		T.120 – International Telecommunications Union (ITU) T.120 contains a series of communication and application protocols and services that provide support for real-time, multipoint data communications. These multipoint facilities are important building blocks for collaborative applications, including desktop data conferencing, and multi-user applications. http://www.imtc.org/t120body.htm
		H323 – International Telecommunications Union (ITU) H.323 addresses Video (Audiovisual) communication on Local Area Networks, including Corporate Intranets and packet-switched networks generally. http://www.imtc.org/h323.htm

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Business Intelligence	Data Management	XBRL – eXtensible Business Reporting Language Extensible Business Reporting Language (XBRL is an open specification which uses XML-based data tags to describe financial statements for both public and private companies. http://www.xbrl.org/
		JOLAP - Java Online Analytical Processing JOLAP is a Java API for the J2EE™ environment that supports the creation and maintenance of OLAP data and metadata, in a vendor-independent manner. http://www.jcp.org/jsr/detail/69.jsp
		XML for Analysis – Microsoft, Hyperion, SAS XML for Analysis uses the Simple Object Access Protocol (SOAP) to let Web browser-based programs access back-end data sources for data analysis. The specification allows companies to build online analytical processing (OLAP) and data mining applications that work over the Web. http://www.microsoft.com/data/xml/XMLAnalysis.htm

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Analytic and Operational Databases	Data Management	JDBC™ – Java™ Data Base Connectivity JDBC™ provides access to virtually any tabular data source from the Java™ programming language. It provides cross-DBMS connectivity to a wide range of SQL databases, and other tabular data sources, such as spreadsheets or flat files. http://java.sun.com/products/jdbc/
		ADO.Net - Microsoft ADO.NET is the data-access component of the Microsoft's .NET Framework. It provides an extensive set of classes that facilitate efficient access to data from a large variety of sources, enable sophisticated manipulation and sorting of data http://support.microsoft.com/default.aspx?xmlid=fh%3BEN-US%3Badonet

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Common Services	Network Services	SNMP V3 – Simple Network Management Protocol SNMP V3 eliminates several of the security vulnerabilities in earlier version. http://www.ietf.org/rfc/rfc2570.txt?number=2570
		LDAP V3 (RFC 1779) – Lightweight Directory Access Protocol Lightweight Directory Access Protocol (LDAP) is a subset of X.500 designed to run directly over the TCP/IP stack. LDAP is, like X.500, both an information model and a protocol for querying and manipulating it. LDAPv3 is an update developed in the IETF (Internet Engineering Task Force), which address the limitations found during deployment of the previous version of LDAP. http://www.opengroup.org/directory/branding/ldap2000/x99di.htm
		X.500 – International Telecommunication Union Telecommunication Standardization Sector (ITU) Defines how global directories should be structured. X.500 directories are hierarchical with different levels for each category of information, such as country, state, and city.

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Intra/Inter Enterprise Integration	Data Interchange Services/ Human Computer Interface Services	XML – eXtensible Markup Language XML has emerged as the standard format for web data, and is beginning to be used as a common data format at all levels of the architecture. Many specialized vocabularies of XML are being developed to support specific Government and Industry functions. http://www.w3.org/XML/
		XSLT – eXtensible Stylesheet Language Transform Transforms XML document from one schema into another. Used for data interchange between systems using different XML schema, or mapping XML to different output devices. http://www.w3.org/Style/XSL/
	Data Interchange Services	ebXML – Electronic Business using XML A modular suite of specifications that enables enterprises to conduct business over the Internet: exchanging business messages, conducting trading relationships, communicating data in common terms and defining and registering business processes. http://www.ebxml.org/
		RDF – Resource Description Framework (emerging) RDF provides a lightweight ontology system to support the exchange of knowledge on the Web. It integrates a variety of web-based metadata activities including sitemaps, content ratings, stream channel definitions, search engine data collection (web crawling), digital library collections, and distributed authoring, using XML as interchange syntax. RDF is the foundation for the Semantic Web envisioned by Tim Berners-Lee - an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation. http://www.w3.org/RDF/ http://www.w3.org/2001/sw/

		DAML+OIL - Defense Advanced Research Projects Agency (DARPA) Agent Modeling Language + Ontology Inference Layer (emerging) DAML+OIL is a semantic markup language for Web resources to allow automated systems to understand the meaning of information even when different terms are used for the same concept. It builds on earlier W3C standards such as RDF and RDF Schema, and extends these languages with richer modelling primitives. http://www.w3.org/TR/daml+oil-reference
		SOAP – Simple Object Access Protocol SOAP provides HTTP/XML based remote procedure call capabilities for XML Web Services http://www.w3.org/2000/xp/Group/ http://msdn.microsoft.com/msdnmag/issues/0300/soap/soap.asp
		WSDL – Web Services Description Language WSDL is an XML based Interface Description Language for describing XML Web Services and how to use them. http://www.w3.org/TR/wsdl
	Data Interchange Services	UDDI – Universal Description Discovery and Integration UDDI provides a searchable registry of XML Web Services and their associated URLs and WSDL pages. http://www.uddi.org/about.html
		Java™ WSDP – Java™ Web Services Development Pack The Java™ Web Services Developer Pack (Java WSDP) is an integrated toolset that allows Java developers to build, test and deploy XML applications, Web services, and Web applications. The Java WSDP provides Java standard implementations of existing key Web services standards including WSDL, SOAP, and UDDI. http://java.sun.com/webservices/webservicespack.html

		WS-I – Web Services Interoperability Organization (emerging) WS-I is an open, industry organization chartered to promote Web services interoperability across platforms, operating systems, and programming languages. http://www.ws-i.org/

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Security	Security Services	X. 509 – International Telecommunication Union - Telecommunication Standardization Sector (ITU-T) Certificate Authentication The international standard for the digital certificate authentication that is used for user identification, especially for creation of an electronic document used to prove identity and public key ownership over a communications network.
		FIPS 186 - Digital Signature Standard (DSS) also Draft ANSI X9.30-199x Part 1; and ISO/IEC JTC1/SC27/WG2, Project 1.27.08 Digital Signature with Appendix) The DSS standard specifies a digital signature algorithm (DSA) appropriate for applications requiring a digital, rather than written, signature. The DSA authenticates the integrity of the signed data and the identity of the signatory. The DSA may also be used to prove that data was actually signed by the generator of the signature.
		S/MIME - Secure Multipurpose Internet Mail Extensions Provides a consistent way to send and receive secure MIME data. Based on the Internet MIME standard, S/MIME provides cryptographic security services for electronic messaging applications: authentication, message integrity and non-repudiation of origin (using digital signatures) and data confidentiality (using encryption). S/MIME is not restricted to mail; it can be used with any transport mechanism that transports MIME data, such as HTTP. http://www.ietf.org/html.charters/smime-charter.html

		SSL - Secure Sockets Layer (SSL) An open, non-proprietary protocol for securing data communications across computer networks. SSL is sandwiched between the application protocol (such as HTTP, Telnet, FTP, and NNTP) and the connection protocol (such as TCP/IP, UDP). SSL provides server authentication, message integrity, data encryption, and optional client authentication for TCP/IP connections.
		TLS – Transport Layer Security Standard for the next generation SSL. Provides communications privacy over the Internet. The protocol allows client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery. http://www.ietf.org/html.charters/tls-charter.html
		WS-Security – Web Services Security Describes enhancements to SOAP messaging to provide message integrity, message confidentiality, and single message authentication. These mechanisms can be used to accommodate a wide variety of security models and encryption technologies including X.509, Kerberos, and SAML. http://www.oasis-open.org/committees/wss/ http://www-106.ibm.com/developerworks/library/ws-secure/
		SAML – Security Assertion Markup Language An XML-based framework for exchanging security information expressed in the form of assertions about subjects, where a subject is an entity (either human or computer) that has an identity in some security domain. SAML is expected to play a key role in the Federal-wide E-Authentication initiative, and is supported by both the Liberty Alliance and WS-Security. http://www.oasis-open.org/committees/security/ http://xml.coverpages.org/saml.html

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Privacy	Human Computer Interface/ Security	P3P1.0 – Platform for Privacy Preferences (emerging) The Platform for Privacy Preferences Project (P3P) provides a simple, automated way for users to gain more control over the use of personal information on Web sites they visit. P3P is a standardized set of multiple-choice questions, covering all the major aspects of a Web site's privacy policies. P3P-enabled Web sites make this information available in a standard, machine-readable format. P3P enabled browsers can "read" this snapshot automatically and compare it to the consumer's own set of privacy preferences. http://www.w3.org/P3P/
		Liberty Alliance An alliance formed to deliver and support a federated network identity solution for the Internet that enables single sign-on for consumers as well as business users in an open, federated way. The Liberty Alliance architecture is largely based on SAML. http://www.projectliberty.org/

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Accessibility	Human Computer Interface	Section 508 – Section 508 of the Rehabilitation Act Requires that Federal Agencies' electronic and information technology is accessible to people with disabilities. http://www.section508.gov/ http://www.section508.gov/index.cfm?FuseAction=Content&ID=12
		Web Content Accessibility Guidelines 1.0, W3C Provides guidelines on how to design and develop web pages that meet accessibility guidelines. http://www.w3.org/WAI/ http://www.w3.org/TR/WCAG10/

Model Component	TRM Service	Starter Set Voluntary Industry Standards
Records Management	Data Management	DoD 5015.2-STD – Department of Defense, Design Criteria Standard for Electronic Records Management Software Applications http://www.dtic.mil/whs/directives/corres/html/50152std.htm
		ISO15489 - International Standards Organization Records Management Standard http://www.iso.org/iso/en/ISOOnline.frontpage