

Section 5. ETA Integration Domain

Integration Domain defines the functions that enable communications in a distributed system and defines the tools that improve the overall usability of an existing architecture made up of products from many different vendors on multiple platforms. Integration tools and products allow organizations to share data between disparate systems that do not communicate easily. Integration tools and products have been described as the software “glue” that allows distributed, multi-tiered applications to work in a world of global networks.

The ETA Integration Domain consists of the following topics: Database Integration, Message Integration, Transaction Process Monitor Integration and Services, Application Integration Middleware and Services, Enterprise Service Bus, Service-Oriented Architecture, Instant Messaging and Mashup.

Domain-wide Requirements

The following domain-wide requirements pertain to all topics and components in the Integration Domain.

- INT-R-01** **Security, Confidentiality, Privacy and Statutes.** Agencies shall implement integration applications/ solutions in adherence with all security, confidentiality and privacy policies and applicable statutes.
- INT-R-02** **Software Tools Version/Release Support.** The version/release levels of all integration software tools shall have vendor or equivalent quality level support available.
- INT-R-03** **Planning.** Before acquiring a central integration solution, agencies shall map their present integration sources and uses, and shall develop a plan in consultation with the Virginia Information Technologies Agency (VITA) Integration Competency Center (ICC) for migration to the central integration solution.
- INT-R-04** **Integration Solutions.** Agencies shall use integration solutions that are scalable, extensible, and maintainable.
- INT-R-05** **Defined Interfaces.** Agencies shall carefully define their interfaces and interface business requirements.
- INT-R-06** **Testing Integration Modifications.** Integration tools and services shall be thoroughly tested. Consideration shall be given to the need to maintain a separate environment for testing modifications.
- INT-R-07** **Shared Resource.** Before acquiring integration solutions, agencies shall contact the VITA ICC to determine if similar integration solutions exist that could be a shared resource across several agencies. To reach the VITA ICC, contact the VITA Customer Care Center (VCCC) by phone 1-866-637-8482, or 804-786-3932 in Richmond, or by Email: vccc@vita.virginia.gov or go online: <http://www.vita.virginia.gov/vccc/incident/vcccincident.cfm>

Database Integration

Database tools and products enable applications to communicate with one or more local or remote databases. They do not transfer calls or objects. For example, database integration does not allow for two-way communication between servers and clients. Servers cannot initiate contact with clients, they can only respond when asked. The discussion of database integration is broken into Directory Services, Metadata, Access Services, and related guidance. Guidance information may direct the reader to other domains once they become available.

Directory Services

A directory may be described as a specialized database of lists. Directories serve a wide variety of functions in a computing environment and are used by applications including email, security, and naming services. Directory services are important as tools in the communications process and a decision about directory services is one of the most important foundational decisions an agency can make in planning a distributed architecture and integration strategy. Having a directory strategy is an integral part of promoting interoperability and, location transparency, and lowering future maintenance costs in a distributed environment.

Directory Services Requirements:

- INT-R-08 Directory Services.** Agencies shall employ Lightweight Directory Access Protocol (LDAP)-compliant directory services. This lays the groundwork for uniform decentralized lists that can be aggregated centrally for use by the Commonwealth.

Table INT-S-01: Directory Services Technology Component Standard <i>Reviewed 10-1-2008</i>	
Strategic:	LDAP, DNS & GDS Sun JDAP; MS Active Directory (ADSI)
Emerging:	None
Transitional/Contained:	X.500 DAP
Obsolescent/Rejected:	Novell NDS

Database Metadata Services

Database metadata services are repositories of data about data. The purpose of the metadata repository is to provide a consistent and reliable means of access to data. The repository itself may be stored in a physical location or may be a virtual database, in which metadata is drawn from separate sources. Metadata may include information about how to access specific data, or more detail about it, among a myriad of possibilities.

Technology Component Standard INT-S-02 provides technology ratings for database metadata services. In general, the technologies listed as strategic are based on open standards.

Table INT-S-02: Database Metadata Services Technology Component Standard <i>Reviewed 10-1-2008</i>	
Strategic:	OMG's UML, MOF MDC's XMI (XML, DTD, Schema) OIM's exchange format XIF (XML) Accessible, computer aided metadata documentation (e.g., ERwin modeling tool) and a metadata repository
Emerging:	Active metadata repository
Transitional/Contained:	Configurable metadata separate from application but proprietary to system.
Obsolescent/Rejected:	Business rules and meaning hard coded into applications. Hard copy only documentation of metadata.

Database Access Services

Database access services refer to software applications that are designed to arrange and store data for ease and speed of search and retrieval.

Table INT-S-03: Database Access Services Technology Component Standard <i>Reviewed 10-1-2008</i>	
Strategic:	DB Adapters or Drivers: ODBC, JDBC, xDBC, OLE-DB (platform specific) XML point to point contracts (e.g., for Schemas) ODBC/SQL compliant gateways XML messaging
Emerging:	None
Transitional/Contained:	OLE (replaced) Screen Scrapers as a mainframe access Non-ODBC/SQL compliant Gateways Translators for non-standard SQL, XML, etc.
Obsolescent/Rejected:	None

Message Integration

Message-Oriented Middleware also known as Message Brokers, MOM, and Messaging Broker, provides an interface between applications or application parts, allowing for the transmission of data back and forth intermittently. Messaging middleware is similar to an e-mail system that transfers messages between people, except that it sends information between applications. MOM

is typically asynchronous and peer-to-peer, but most implementations support synchronous message passing as well. In general, a message-oriented middleware has one of two architectures: the hub-and-spoke model or the network-centric bus model, also called the message-bus model. If the destination application is not available because of connection failure or because the application is busy, the middleware stores the data in a message queue until the application becomes available.

Message Formats

In this section, the term “messages” will be used in the broadest sense to encompass transaction-based messages as well as entire file transfers. To many messaging systems, the format of the content of the message doesn’t matter as long as it has the understood envelope/wrapper or an operating system recognizable format. However, the format of the content is very important to the receiving operating system, application, or user. Format translations may be performed by integration products. Also included in this section are messages that are object-oriented. These messages are requests or replies that are issued or received by applications or databases.

Table INT-S-04: Message Formats Technology Component Standard <i>Reviewed 10-1-2008</i>	
Strategic:	XML and CSS (presentation style configurable by administrator for device types) 7 bit ASCII; 8 bit ASCII; EBCDIC (translation)
Emerging:	None
Transitional/Contained:	None
Obsolescent/Rejected:	None

Message Transfers

Message transfers refer to software applications that are designed to provide for correct and reliable end-to-end data transport between communication partners.

Table INT-S-05: Message Transfers Technology Component Standard <i>Reviewed 10-1-2008</i>	
Strategic:	File and Data Requests/Replies FTP XML file transfer Presentation and Translation Services for Security Encryption/Decryption Services (A wide variety of encryption algorithms are strategic depending on security needs) e.g., Symmetric Encryption, DES, Triple DES, RC2, RC4 Terminal Emulation APPC LU6.2
Emerging:	None
Transitional/Contained:	Presentation and Translation Services for Security Proprietary style layout separate from application Terminal Emulation SNA/SDLC (OSI level 2)
Obsolescent/Rejected:	FTP whenever security required

Messaging Integration

The recommended messaging protocols also know as email (electronic mail) protocols apply to mail messaging and/or other application-to-application messaging. Email is the exchange of computer-stored messages by telecommunication. Mail programs should support use of MIME (Multipurpose Internet Mail Extensions), be SMTP/ESMTP enabled (Simple Mail Transfer Protocol/Extended Simple Mail Transfer Protocol), and provide proxy through IMAP4/POP3 servers (Internet Message Access Protocol 4/Point of Presence 3). Mail programs that interface with Windows clients use Microsoft's MAPI (Messaging Application Programming Interface) interface. Middleware protocols used by mail applications and/or other applications include: LDAP, DNS (Domain Name System), SSL (Secure Sockets Layer), and additional security protocols.

Message Integration requirements

- INT-R-09** **Email Protocols.** Agency email messaging shall be SMTP and MIME compatible. Local governments are encouraged to follow this standard as well.
- INT-R-10** **Emails.** The Message Transfer Agent (MTA) in email applications should be LDAP enabled.

Table INT-S-06: Message Integration Technology Component Standard <i>Updated 10-1-2008</i>	
Strategic:	IMAP MAPI SMTP/MIME XSL (presentation style and content configurable by user)
Emerging:	XSL (presentation style and content configurable by user)
Transitional/Contained:	X.400 POP3 VIM CMC
Obsolescent/Rejected:	Non-Internet compatible email

Transaction Process Monitor Integration and Services

Distributed transaction processing ensures transaction integrity for transactions that involve databases. Transaction processing is the independent execution of a set of operations on data in a relational database, which treats that set of actions as a single event. If any part of the transaction process fails, the entire transaction fails and all participating resources are rolled back to their previous state.

Transaction processing monitors and some web services software are critical to the 3-tier application client/server computing model because they facilitate writing of the programs that track transactions across multiple platforms. In the n-tier world, the application layer functions between the presentation layer on the PC and the data layer on the mainframe, Unix, or Windows-based systems. Historically some of the following services have been included in transaction processing monitor middleware: two-phase commits, failure/recovery, synchronization, scheduling, repeat attempts, business-rule-based transaction workflow services, message queuing resource managers, and load balancing. Perhaps the most significant feature of the TP monitor is its ability to funnel database requests.

Technology Component Standard INT-S-07 provides strategic open protocols and examples of mainframe programs used to define the typical work performed by transaction processing monitors. In general, those technologies listed as strategic are based on open standards.

Table INT-S-07: Transaction Process Monitor Integration and Services Technology Component Standard <i>Updated 10-1-2008</i>	
Strategic:	SOAP WSDL HTTP M-POST
Emerging:	None
Transitional/Contained:	X/Open: XA interface (X/Open is the standard, XA is the interface) STDL (structured transaction definition language) DTP (distributed transaction processing) CPI-C (common program interface for communications) CORBA DCOM
Obsolescent/Rejected:	None
Historical Note: Two TP monitors were widely used in the mainframe world and then later transitioned to the client-server world. These were CICS (customer information control system) and ACMS (automated code management system).	

Application Integration Middleware Servers and Services

Application integration middleware provides interfaces to a wide variety of applications. Application integration middleware might be a service that enables running a legacy system through a thin-client browser or a service that enables the execution of multiple application functions from an integrated user interface. The methods used to achieve this integration include application program interfaces (API), remote procedure calls (RPC), and object request brokers (ORB).

Protocols and services related to application integration are noted in Technology Component Standard INT-S-08. In general, those technologies listed as strategic are based on open standards.

Table INT-S-08: Application Integration Services Technology Component Standard <i>Updated 10-1-2008</i>	
Strategic:	Object Request and Request Broker Protocols/Suites .NET Remoting SOAP over HTTP J2EE/RMI, Java 2 Enterprise Edition (the distributed version) and Remote Method Invocation Enterprise Application Integration Services (EAI) Use of Integration Servers/Services SOA Remote Procedure Calls DCE RPC DCE secure RPC (integrated with DCE security protocols for authentication, protection level and authorization) Web Services Object and Application Interfaces IDL (interface definition language) stubs; MIDL (Microsoft); OMG IDL; DCE IDL
Emerging:	None
Transitional/Contained:	Remote Procedure Calls Suns' ONC+ RPC MS DCOM + (distributed common object model) OMG CORBA (common object request broker) DCE RPC DCE secure RPC (integrated with DCE security protocols for authentication, protection level and authorization) ebXML
Obsolescent/Rejected:	None
Historical Note: Fully utilizing Web Services is the recommended strategic direction when combined with an overall Service-Oriented Architecture. For a description of SOA please see Appendix A of the ETA Application Domain Report, <i>Example SOA Centralized Implementation and Governance Model</i> . Other methods, such as DCOM and CORBA are still used and recommended for specific scenarios.	

Enterprise Service Bus

An enterprise service bus (ESB) is a Web-services-capable middleware infrastructure that supports communication and mediates application interactions. To be an ESB, a middleware subsystem must

1. implement program-to-program communication (always supporting Simple Object Access Protocol/Hypertext Transfer Protocol [SOAP/HTTP], and almost always supporting SOAP on message-oriented middleware [MOM] and plain MOM);

2. support other Web services standards (including Extensible Markup Language [XML] and Web Services Description Language [WSDL]);
3. be capable of service discovery, binding and virtualization (transparently substituting alternative service providers) and intelligent message routing;
4. have an extensible, intermediary-based architecture so that additional features can be plugged in; and
5. have an awareness of message schemas through the use of metadata. ¹

Instant Messaging

Instant Messaging² is the exchange of text messages through a software application in real-time. Generally included in the IM software is the ability to easily see whether a chosen friend, co-worker or "buddy" is online and connected through the selected service. Instant messaging differs from ordinary e-mail in the immediacy of the message exchange and also makes a continued exchange simpler than sending e-mail back and forth. Most exchanges are text-only, though popular services, such as AOL, MSN Messenger, Yahoo! Messenger and Apple's iChat now allow voice messaging, file sharing and even video chat when both users have cameras.

Products and services related to instant messaging are noted in Technology Component Standard INT-S-09. In general, those technologies listed as strategic are based on open standards.

Table INT-S-09: Instant Messaging Technology Component Standard <i>Added 10-1-2008</i>	
Strategic:	IBM Lotus Sametime Jabber XCP Microsoft Live Communications (Server/Office Communication Server)
Emerging:	Bantu EIM Parlano MindAlign Sun Microsystems Java System Instant Messaging
Transitional/Contained:	Novell GroupWise Messenger
Obsolescent/Rejected:	None

¹ Integration Suites and ESBs: Integration Technology for the Mainstream. Jess Thompson & Roy Schulte. Gartner Research.

² Wikipedia, April 2008: http://en.wikipedia.org/wiki/Instant_Messaging

Mashup

A “mashup”^{3, 4} is a lightweight, tactical presentation layer integration of multi-sourced applications or content into a single, browser-compatible offering. Mashups⁵ currently come in three general types: consumer mashups, data mashups, and business mashups.

Mashups⁶ leverage content and logic from other Web sites and Web applications, and are built with a minimal amount of code (which can be client-side JavaScript or server-side scripting languages, such as PHP or Python). Mashups aren't intended to be strategic, systematically built, industrial-strength enterprise applications; rather, they're created quickly or opportunistically to meet a focused tactical need. Mashups are generally personalized to fulfill personal productivity needs rather than the requirements of a long-standing corporate role.

Protocols and services related to mashups are noted in Technology Component Standard INT-S-10. In general, those technologies listed as strategic are based on open standards.

Table INT-S-10: Technology Component Standard <i>Added 10-1-2008</i>	
Strategic:	<p>Ajax - AJAX (Asynchronous JavaScript and XML) is a group of interrelated web development techniques used for creating interactive web applications.</p> <p>EDA - Event-driven architecture</p> <p>SOA - Service-oriented architecture</p> <p>WOA - Web-oriented architecture</p> <p>URI - Uniform resource identifiers</p> <p>Rest - Representational state transfer</p> <p>ATOM - the Atom Publishing Protocol is a simple HTTP-based protocol for creating and updating web resources.</p> <p>RSS - RSS (Really Simple Syndication) is a family of Web feed formats used to publish frequently updated content such as blog entries, news headlines or podcasts.</p> <p>Use available API's wherever possible http://www.programmableweb.com/apis/directory/1?sort=mashups</p>
Emerging:	None
Transitional/Contained:	None
Obsolescent/Rejected:	None

³ The source for much of the information presented in the Mashup sections was obtained through Gartner Research, Gartner, Inc. Stamford, CT.

⁴ Anthony Bradley, Daniel Sholler, David Gootzit. *Enterprise IT Departments Must Prepare for the Impact of “Mashups”* 7 September 2007 Gartner Research: ID G00151424 Retrieved November 2007.

⁵ Wikipedia: http://en.wikipedia.org/wiki/Mashup_%28web_application_hybrid%29. Retrieved November 2007.

⁶ Wikipedia: http://whatis.techtarget.com/definition/0,,sid9_gci1167147,00.html. Retrieved December 2007.