SOA Design Patterns for VistA Evolution: COTS Applications

Office of the Chief Technology Strategist (TS)

Architecture, Strategy, and Design (ASD)

Office of Information and Technology (OIT)

Version 1.2

Date Issued: 15 April 2014

THIS PAGE INTENTIONALLY LEFT BLANK FOR PRINTING PURPOSES

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Purpose

The purpose of this document is to provide the VA Enterprise Design Pattern for the development and implementation of VistA Evolution applications in a services oriented manner. Enterprise design patterns will provide the technical strategy and implementation guidance required for VA to align to best practices, standards, and guidance that constitute the OneVA Enterprise Architecture (EA).

The design patterns will be leveraged by VA to inform and constrain program-specific solution architectures. Solution architecture represents detailed product configurations and interface specifications, and guide full-lifecycle system design, integration, testing, and deployment processes for individual programs. Enterprise design patterns provide strategic guidance to assist programs in adopting the appropriate technologies to align with the VA Enterprise Technology Strategic Plan (formerly the VA IT Roadmap).

1.2 Scope

This design pattern document applies to all VistA Evolution COTS (including open-source and GOTS software) applications as part of the VistA Evolution family of systems. It is intended for all new COTS healthcare applications that integrate into the VA enterprise architecture and share data with VistA, regardless of end-user device (device-independent). These will apply to both new applications as well as modified systems currently in production.

1.3 Document Development and Maintenance

Developed collaboratively with stakeholders from OIT Product Development (PD), Office of Information Security (OIS), Architecture, Strategy and Design (ASD), and Service Delivery and Engineering (SDE), design patterns will guide and synchronize the development of system designs to drive the realization of a common technology vision for the enterprise. This vision, which is documented in the VA Enterprise Technology Strategic Plan, leverages best-of-breed technologies to maximize the effectiveness, efficiency and security of the VA's IT assists. This creates a feedback loop which cultivates participation and collaboration between both enterprise architecture and solution architecture domains.

This document will be reviewed and updated as needed to account for additional feedback from stakeholders as well as lessons learned from enterprise design pattern implementation. Updates will be coordinated with the Government Lead for this document, who will facilitate stakeholder coordination and subsequent re-approval. Major updates of this document will require formal re-approval per the approval chain listed in the "Approval Coordination" section.

2 DESIGN PATTERN OVERVIEW

A service-oriented architecture (SOA) is an application topology in which the business logic of the application is organized in modules (services) with clear identity, purpose and programmatic-access interfaces. Services behave as "black boxes" where their internal design is independent of the nature and purpose of the requestor. In a SOA, data and business logic are encapsulated in modular business components with documented interfaces. This clarifies

design and facilitates incremental development and future extensions. A SOA application can also be integrated with heterogeneous, external legacy and purchased applications more easily than a monolithic, non-SOA application can.

2.1 Business Need

VA is planning for the evolution of the Veterans Information Systems and Technology Architecture (VistA) from a set of decentralized legacy systems to an integrated, modern SOA environment. This evolution requires strategic architecture guidance to facilitate program-level decisions about design approaches that will support the modernization of VistA applications. This pattern is part of a library being developed for VistA Evolution, and addresses the following use case: COTS applications that share data with other VA applications and use VA SOA infrastructure services as appropriate.

2.2 VistA Evolution Architectural Approach

As shown in Figure 1, VistA Evolution will be implemented using a SOA-based approach. This approach environment will leverage a combination of enterprise application and SOA support infrastructure services (including Identity and Access Management), as explained in the following section.
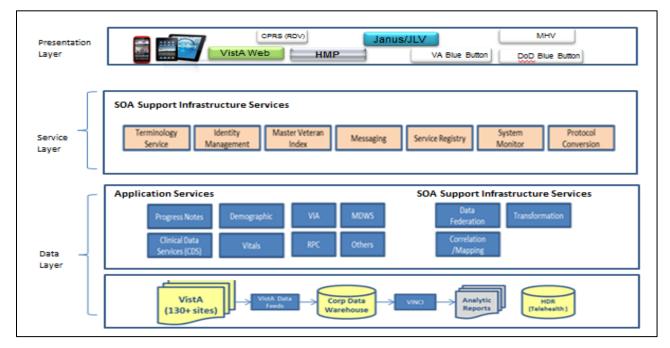

Figure 1. VistA-4 IOC Technical Architecture Concept

3 DESIGN PATTERN DESCRIPTION

COTS applications (including open-source solutions) which are part of VistA Evolution will use common enterprise services (Enterprise Common Services – see Appendix B) to facilitate re-use, achieve economies of scale, and to reduce development and maintenance costs. This document defines these services in two separate categories as follows:

- Application Services
  - CRUD (Create, Read, Update, and Delete) data services (e.g. direct data access services involving CRUD operations for service consumers)
  - Composite data services (e.g. may include a composition of functions that provide data manipulation or to provide aggregate responses to service consumers from multiple data sources)
- SOA Support Infrastructure Services, which include (but are not limited to):
  - Messaging/Enterprise Service Bus (ESB) (e.g. message exchange transport, service description and discovery, XML parsing)
  - Enterprise SOA infrastructure services (e.g. end-to-end application monitoring, authentication, authorization, auditing, event management, orchestration)

The following sections explain specific technical attributes that COTS applications interfacing with VistA will take into account to align to the VA's long-term technology strategic vision.

3.1 Rules

VistA applications generally have a presentation layer at the top, a services layer in the middle, and a data services layer at the bottom. These applications will use a mixture of application and SOA support infrastructure services to provide information to end users. The following concept diagram shows an example of how the to-be VA SOA can be applied to different types of VistA applications using a combination of application and infrastructure services.
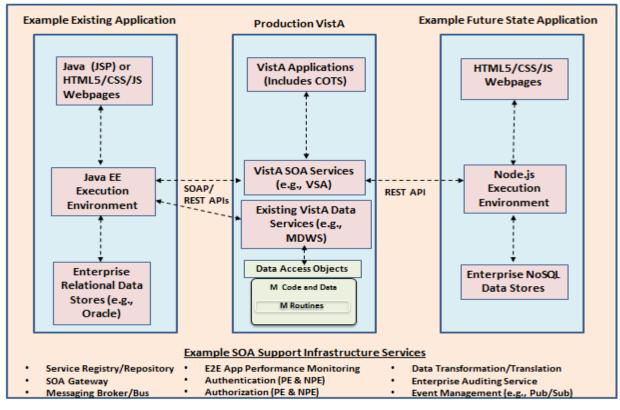

Figure 2. SOA Environment Applied to VistA Using a Mix of Application and Infrastructure Services

The top portion represents a user application layer, which includes dynamic webpages written in HTML5, Cascading Style Sheets (CSS) and JavaScript, or a COTS application (e.g., laboratory

application). These will be abstracted from business logic in an execution environment (e.g., Java Platform Enterprise Edition, Node.js) that is exposed by web services and mediated by SOA infrastructure services to exchange data to end users. Service consumers use data services that provide virtual access to enterprise data stores such as VistA, traditional relational SQL databases, or persistent NoSQL data stores in enterprise instances of shared data (see Section 5.3). Applications may fall into three categories for which this design pattern may apply:

1. Existing COTS applications that are developed in traditional development environments and access VistA via a combination of composite and CRUD services. These applications may leverage existing VistA services, such as Medical Domain Web Services (MDWS) (to be retired in the to-be VistA SOA environment) or new VistA SOA services developed by the VistA Service Assembler (VSA) toolkit (currently under development).
2. Production VistA data services that currently use virtual data access layers, and are also currently being enhanced by exposing SOA services using a toolkit such as the VistA Service Assembler (VSA) toolkit to wrap existing MUMPS code (aka M code) and associated technology investments. These investments currently include VistA middleware currently in production (e.g., Cache objects).
3. New future-state "green field" applications developed in a modern scripting language framework that use emerging open-source toolkits and lightweight web server technologies (e.g., Node.js), using a mix of composite and CRUD services to access VistA data while leveraging VA infrastructure services.

To develop new applications for any of the above scenarios, the following rules must be taken into account in order to integrate into the VistA SOA environment:

1. Follow open nationally approved standards for data sharing (e.g., SNOMED, HL7) as approved by the VA
2. No modifications to COTS applications will be made unless the vendor agrees to include modifications in the general release of that product
3. Application functionality shall be exposed to VA to be available as SOA services.
   a. In accordance with current Technical Reference Model (TRM) guidance for messages sent over HTTP as the transport protocol
   b. Provides asynchronous and stateless services using the REST architectural style
4. Projects implementing COTS applications shall leverage the TRM and OneVA Enterprise Technical Architecture (ETA) compliance criteria.
5. COTS applications shall interface with VA enterprise common services as approved by VA and listed at the Enterprise Shared Services (ESS) Website
6. No new direct system-to-system interfaces will be developed or fielded
7. RESTful web services using Hypermedia as the Engine of Application State (HATEOAS) will leverage an ESB for service call load balancing and endpoint management

3.2 Application Services

Application services consist of web services to provide access to data sources, either via atomic data services or using a composition of atomic services to manipulate data from disparate data

sources and providing an aggregated response. The following concept diagram shows the scope of application services in the VistA SOA environment:
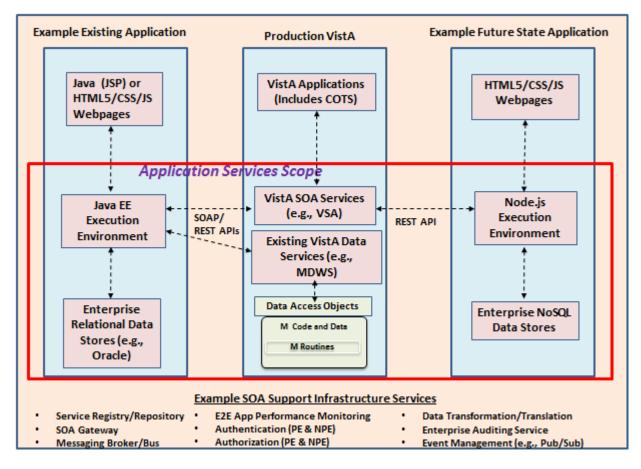

Figure 3. SOA Environment Applied to VistA - Application Services

The following technical attributes pertain to application services in the to-be VistA SOA environment:

1. No new direct system-to-system interfaces will be developed or fielded
2. We will maximize the VA investment in existing MUMPS functionality by exposing them as web services
   a. Aggregation Services will be leveraged to re-use existing composite services
   b. Existing MUMPS services will be exposed as RESTful web services
3. All new services will be developed in widely used languages (e.g., Java, JavaScript, PHP) and will leverage open-source development toolkits to develop services that interface with VistA data services
4. VistA CRUD Services (including those implemented by the Retrieve, Locate, and Update Service (RLUS) API)
   a. VA will drive toward the minimum possible number of service providers for each CRUD service interacting with VistA (directly or indirectly)

b. All VistA CRUD operations will leverage "contract first" development to abstract and separate/protect consumers against service provider back-end changes

c. All VistA CRUD operations will be developed in accordance with service contracts based on industry standards and identified in enterprise service registries

5. All composite services will be exposed as RESTful web services in the to-be SOA environment

   a. SOA infrastructure services will be responsible for ensuring that all subscribers receive information to which they subscribe in the format (e.g. JSON vs. XML, etc.) that they require.

   b. Service providers will not be required to provide information in multiple message formats

   c. Any and all enterprise level services will use an officially designated SOA infrastructure services

   d. Any service that will be invoked across security boundaries must use the SOA infrastructure services, and be exposed using a service proxy (which may include a service façade or adapter).

   e. When using SOA infrastructure services, it will provide the service proxy The SOA infrastructure service shall not be used by clinical decision support or messaging between medical devices on a case by case basis

   f. Quality of Service (QOS): Service consumers accessing data via SOA infrastructure services must satisfy QOS requirements per the Service Level Agreements (SLA) of the service providers.

   g. The SOA infrastructure service, which contains service registry information including the SLAs, must implement QOS handling.

   h. Service providers will not be required to deal directly with QOS.

      i. REQUIREMENT: The network control devices shall be configured to support QOS requirements

6. Persistence

   a. VistA applications will use officially designated enterprise services for data persistence that include:

      i. Persistence of data at the application level made available to the enterprise as a web service

      ii. Persistence of data at the SOA infrastructure level made available as a publish/subscribe service

   b. New capabilities that require persistence must use standard enterprise-level CRUD data services providers

      i. The CRUD services will ensure that data that is captured locally (caching) will also persist the data at the enterprise level (logical versus physical store)

   c. Required enhancement to the legacy interfaces will be made as necessary to support clinical use of new capabilities

   d. For new data sets, there are three persistence options from which a PM must select the most appropriate:

      i. New data sets that can be mapped to existing VistA data will be persisted both at the enterprise and the local level

      ii. New data types/sets that do not exist in VistA will be persisted at the enterprise level versus locally. NOTE: This does NOT include enhancements to existing VistA data sets such as additional immunization data from a new source

      iii. For some extensive enhancements to the existing data set, it may make sense to modify the legacy Computerized Patient Record System (CPRS) interface to present data pulled from both the local VistA data store and at the enterprise shared data instances (NoSQL and SQL)

7. Application/Data services (CRUD and Composite)
   a. No application will access data directly without using an officially designated enterprise CRUD and/or Composite services
   b. Data is not useful without context. Therefore, the service call must include a reference to a source for providing context to the data (e.g. XML Schema, etc.)
      i. *REQUIREMENT: VA must adopt the/a industry standard for indicating the provenance of a document*
   c. If passing a coded medical term, the code system must go with it.

3.3 SOA Support Infrastructure Services

SOA Infrastructure services are not directly part of the application itself, but may be used by applications that utilize the SOA infrastructure. The following figure shows the scope for the infrastructure services that fit into the VistA SOA environment:
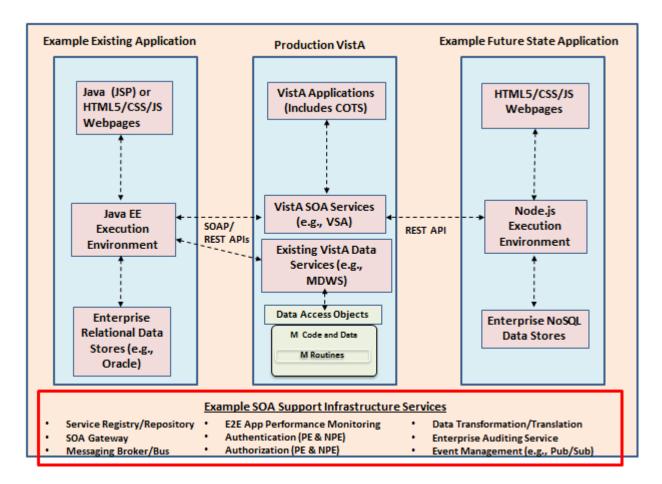
Figure 4. SOA Environment Applied to VistA – SOA Support Infrastructure Services

These services are enterprise-level and perform important duties such as message brokering, authentication, authorization, and performance monitoring for all applications that are integrated into the SOA. To achieve this vision, the following attributes must be taken into account:

1. COTS applications shall:
   a. Have an interface specification/service that enables extraction of audit information and aggregation of disparate audit information from multiple sources
   b. Leverage an enterprise-wide audit system that complies with NIST, VA, and DoD standards
      i. *REQUIREMENT: VA must implement an enterprise audit shared service.*
2. SOA service providers shall generate and pass to the service consumer a meaningful error message that allows the service consumer to decide on fault mitigation. The service provider with which the error originates must log the error to an enterprise logging service.
      i. *REQUIREMENT: VA must implement an enterprise error and exception shared service.*
3. SOA support infrastructure services will include end-to-end application performance monitoring with which COTS applications must be compatible
4. Authentication for PE

a. All COTS applications shall use the VA enterprise service for Direct Client Authentication using Public Key Infrastructure (PKI) over Transport Layer Security (TLS) (authentication is passed) for internal users (provided by the VA Identity and Access Management (IAM) program)
   i. *ASSUMPTION: If a risk assessment requires LOA4 then direct client authentication using PKI over TLS is needed. If LOA2 or LOA3 PKI authentication to an IAM SSO would be acceptable.*
b. Using a VA approved authentication methodology per NIST 800-63, VA Handbook 6500, and ESS Security Model guidance
c. Authenticate (and subsequently authorize) the requestor in the presentation layer prior to application invocation and authentication information will be included as an integral part of the request message
d. All privileged user accounts will use special administrator tokens
e. In the near-term, VistA Kernel authentication will be enhanced to integrate with enterprise level authentication, authorization and audit capabilities
f. A SOA gateway will use authentication contained in the message to permit the information request to cross the gateway

5. Authentication for NPE
   a. For services requests from user interface to VistA/VA service provider: System authentication will be provided by an enterprise NPE authentication service
   b. For responses from VistA/VA service to user interface: System authentication will be provided by an enterprise NPE authentication service

6. PE Authorization (After Authentication)
   a. For the COTS application itself: User authorization will be provided by either:
      i. Internal (consume authenticated user attribute data from the authentication) to determine authentication/access control, or,
      ii. Enterprise ABAC/RBAC service (e.g., use of XACML for authorization)
   b. For service requests from COTS user interface to VistA/VA Service provider: User authorization will be provided by enterprise user ABAC/RBAC service
   c. For service requests, user authorization will be provided by an enterprise ABAC/RBAC service
      i. *ASSUMPTION: Fine grained authorization attributes could be passed in or obtained from Policy Information Points (PIP) including Identity stores, Provisioning Stores, Clinical Data Stores, and Environment*
   d. The enterprise service must enable approved user attributes to be managed locally; enterprise attributes and authorization rules are managed at the enterprise level and cannot be changed at the application or local level

7. NPE Authorization
   a. For services requests from COTS to VistA/VA Service provider: System authorization will be provided by enterprise NPE authorization service (PKI issued by a common root certificate)
   b. For services requests from VistA/VA Service to COTS service: System authorization will be provided by enterprise NPE authorization service

    c. For services requests from external partners making system to system service calls to VA COTS or VistA systems: Authorization will be executed in accordance with industry (ONC, HL7) standards

8. Enterprise Data Mediation/Terminology Service
    a. VistA needs to make the internal enhancements to make use of industry standards; service will always be needed at a minimum to address versions as they evolve.
    b. VA will work with standards organizations to develop and propose new data sharing standards depending on programmatic needs.

9. Transformation /Translation
    a. Use existing SOA infrastructure services for sending a message that requires transformation or translation
    b. Messages that require transformation or translation will use existing SOA infrastructure capabilities to the maximum extent appropriate, applicable and available
        i. If such capabilities are not available they should be developed as part of the SOA infrastructure (and made available to OSEHRA and meet their standards).
    c. If the application includes a capability that is embedded as part of another system, then the capability should be exposed as a SOA service.
    d. The General Interface pattern will have four parameters:
        i. What needs to be translated;
        ii. The content/format from which it needs to be translated from;
        iii. The content/format that it needs to be translated to;
        iv. Offer options. There may be additional parameters that may be taken into account

10. Aggregation
    a. All aggregation services must leverage appropriate/applicable CRUD services; No aggregation service shall directly access data
        i. Specifically, DoD data will be accessed via a service and not via a direct interface to the DoD data store. The same also applies to VistA instances.
    b. No application shall build a new aggregation services. Aggregation services shall be provided by SOA infrastructure services

11. SOA Gateway
    a. No application shall implement a new gateway to infrastructure services. The gateway will be provided with existing enterprise SOA infrastructure services.

12. Registration and Discovery
    a. All SOA services will be registered at design time in an enterprise service registry and repository, and will be used by applications at both design time and run time to select the appropriate service providers to satisfy consumer needs

4 PE AUTHENTICATION AND AUTHORIZATION FOR VA CREDENTIALED USERS

4.1 Internal COTS Application Functionality

Additional attributes for authentication and authorization for VA credentialed users accessing internal COTS application functionality are shown in the following figure:
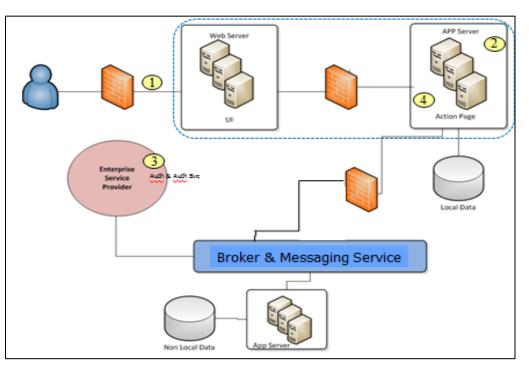

Figure 5. Internal COTS Functionality Authentication and Authorization

This authentication and authorization process consists of the following aspects:

1. Client presents PKI credentials (via PIV cards) over TLS connection to COTS application
2. COTS application redirects client to enterprise authentication service
3. Enterprise Service authenticates user's identity and returns PKI based token with user attributes
4. COTS applications uses PKI token (e.g. SAML) and embedded attributes to make access control decisions

The COTS application maintains session security in accordance with enterprise guidelines. Attributes may include "roles" for the purpose of RBAC, and TLS is only between the client and web server.

4.2 VistA Service Functionality

Attributes for authentication and authorization for VA credentialed users Accessing VistA Functionality via a Service is shown in the following figure:
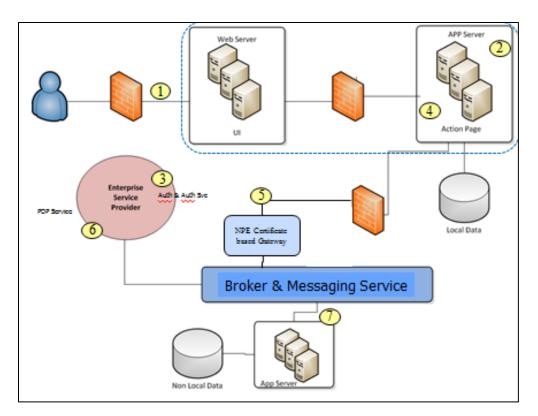
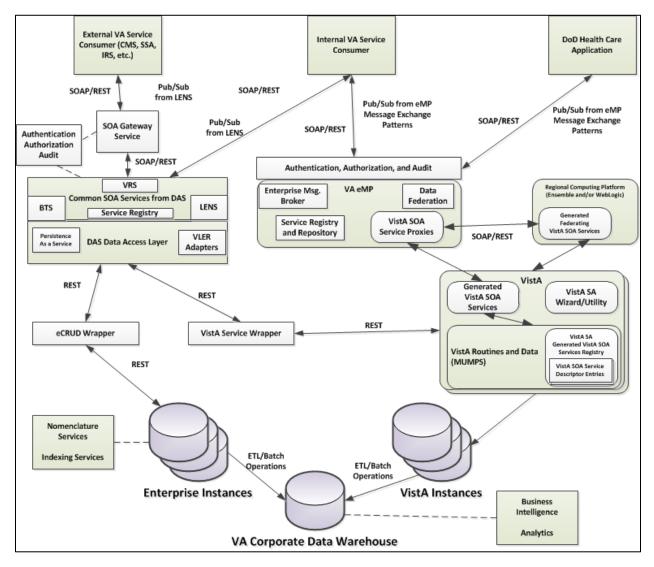**Figure 6. VistA Service Functionality Authentication and Authorization**

This process consists of the following aspects:

1. Client presents PKI credentials over TLS connection to COTS application
2. COTS application redirects client to enterprise authentication service
3. Enterprise Service authenticates user's identity and returns PKI based token with user attributes
4. Application use brokered authentication token (e.g. SAML) and embedded attributes to make access control decisions
5. Application establishes connection to broker via application gateway (Encryption ends at gateway)
6. Broker forwards request information to enterprise service to determine if user and/or application is authorized to access the requested data
7. If yes, Broker forwards request to data service provider with unaltered user and consuming application credentials
8. If no, Broker returns appropriate denial message

Clinical provider information associated with provenance of medical information is captured in the message payload as dictated by ONC and industry standards.

5 ENTERPRISE SERVICES VISION FOR VISTA EVOLUTION

The following notional architectural diagram shows how diverse applications internal and external to the VA will be able to use a combination of enterprise services provided by VLER DAS and the VA Enterprise Messaging Infrastructure (eMI) or Enterprise Messaging Platform (eMP) (formerly SOA Suite or Enterprise Service Bus) to share data through loosely coupled SOA-based services.

Figure 7. Use of Enterprise Services by Applications that Access VistA (Conceptual Overview)

This figure also includes references to enterprise data store instances, which include enterprise NoSQL and SQL data stores for persistent data, and it also incorporates an enterprise data warehouse capability that may be used for other applications such as analytics and business intelligence. Each portion of DAS and eMP contain a set of support services that constitute the SOA infrastructure that applications may use to mediate connections between both consumers and producers of business services.

5.1 VA eMP (formerly SOA Suite)

The following notional architectural diagram shows how diverse applications internal and external to the VA may use enterprise services brokered by the VA eMP to obtain data through SOA-based VistA services.
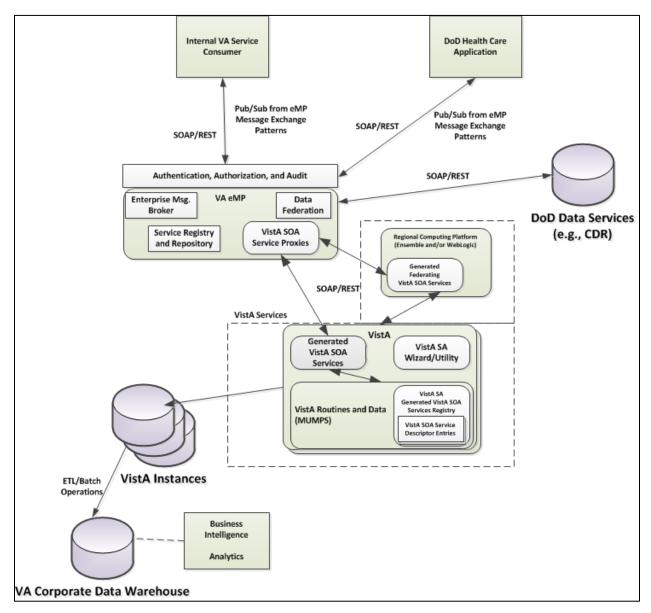

Figure 8. Use of the VA eMP for Applications that Access VistA (NOTIONAL)

The to-be vision for VistA SOA implementation calls for the use of enterprise services to broker message exchanges between service consumers and service providers. The VA eMP provides an Enterprise Service Bus (ESB) that performs message brokering and data federation capabilities for VistA transactions within appropriate user security and privacy parameters. New applications may also use this capability to integrate into end-to-end performance monitoring and analytics services without having to acquire their own application-specific services.

Additionally, the VA eMP integrates seamlessly into VistA data services created with the VistA Service Assembler (VSA) toolkit, as explained in the following section.

5.2 VistA Service Assembler (VSA)

Existing business logic for VistA may be exposed as SOA services using the VSA toolkit as explained in the following figure:
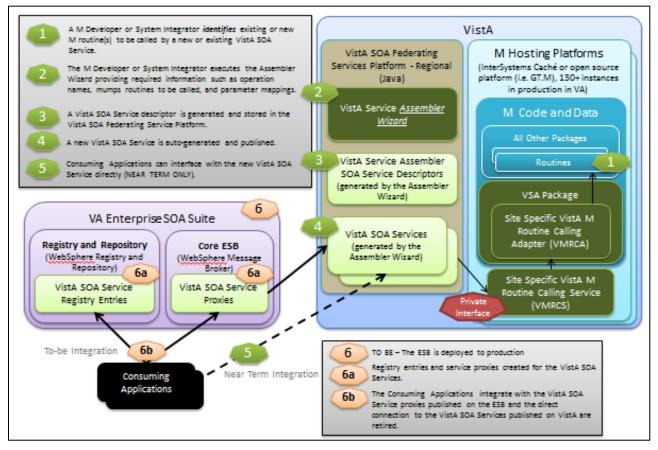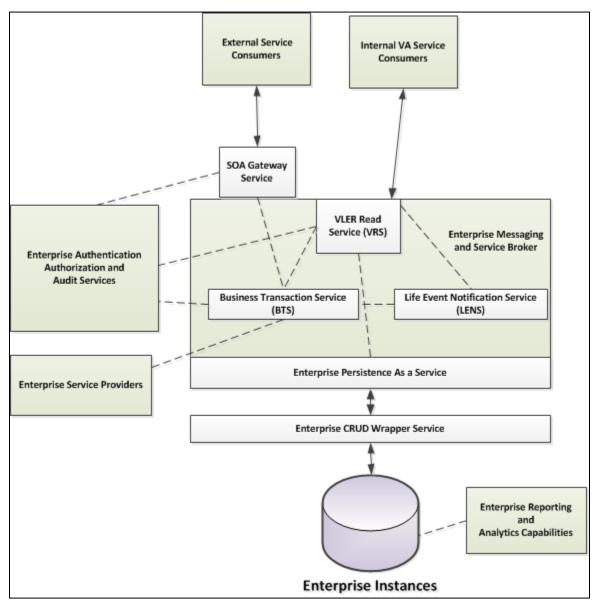

Figure 9. Overview of the VistA Service Assembler (VSA) Toolkit (OIT Product Development)

The VSA toolkit is a web application developed by OIT Product Development (PD) that automates SOA service generation, and it is utilized at design time using developer specified definition. Existing business logic remains in the traditional MUMPS environment (M code), and the VSA-created services facilitate SOA Service payload transformations (such as XML, JSON, etc.) and a federation of multiple VistA systems (one, many, all). It will result in standardized, reliable VistA based SOA services, and will bridge the staff "orientation gap" between traditional MUMPS programming and SOA service creation technologies. VSA will be used to provide all data services that access VistA data, while no new business logic developed in MUMPS will be used in the to-be VistA SOA environment.

5.3 Enterprise Data Persistence

In the future, new applications will be required to leverage officially designated common enterprise services, including enterprise CRUD services and the enterprise instances of shared

data stores (including SQL and NoSQL). The following notional figure illustrates how specific applications integrate with the Virtual Lifetime Electronic Record (VLER) Data Access Services (DAS) architecture to obtain information for service consumers.

Figure 10. Use of Enterprise Shared Data Instances in the To-Be VA SOA Environment (Conceptual Overview)
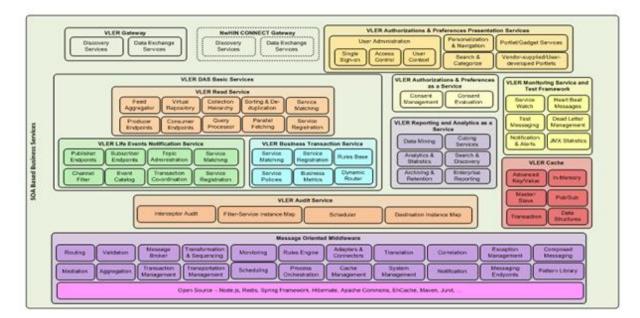
New applications may integrate with DAS as well as other enterprise SOA capabilities, such as the VA eMI, as explained in Section 5.1. More information about the VLER DAS SOA services and data access layers can be found in Appendix A.

APPENDIX A: VLER DAS SOA AND DATA ACCESS LAYERS

The following figure shows the different types of services that are contained within the VLER DAS capability.
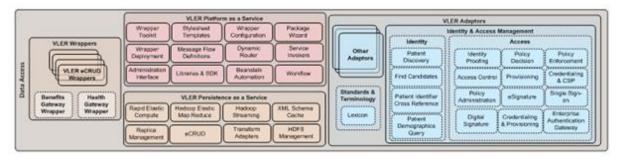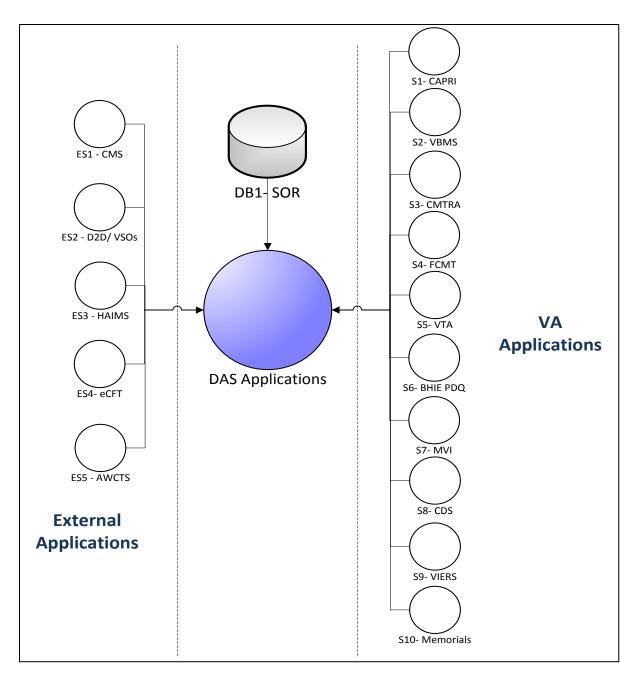

Figure A-1. VLER DAS SOA Services and Data Access Layers (VLER DAS System Design Document – 30 October 2013)

These are intended to be enterprise-grade services that applications will need to use in order to access appropriate data and provide responses to end users. Using these services, VLER DAS will be able to integrate many diverse service consumers and producers into the SOA environment, as shown in the following notional data flow diagram (per the VLER DAS System Design Document (SDD)):

Figure A-2. VLER DAS Data Flow Diagram (VLER DAS System Design Document – 30 October 2013)

It is envisioned by the VA that VLER DAS will develop a set of SOA services that, combined with other capabilities in the VA (e.g., eMP), will provide a common set of shared SOA services that all applications may use to meet their specific business requirements. These services will be abstracted from the application layer so that programs can focus primarily on front-end development and standard interfaces to an integrated, enterprise back-end

APPENDIX B: ENTERPRISE COMMON SERVICES

The following notional figure illustrates where common services may be applied in the VA versus other non-enterprise-grade services that may only be germane to a specific community.
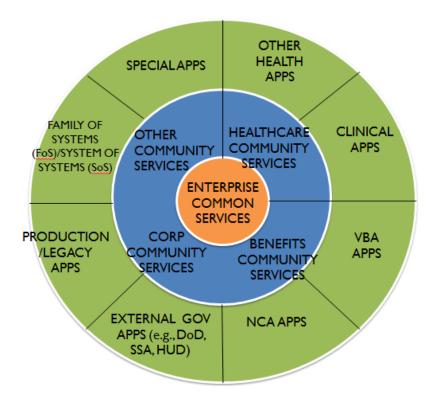

Figure B-1. High-Level Concept of Enterprise Common Services

The outer layer represents the many different types of applications that may use enterprise common services. These applications can be configured to support a wide array of business purposes and customer desires. These services are shared services, which will be accessed via enterprise-wide service registries and referenced on the Enterprise Shared Services (ESS) website, as maintained by OIT Architecture, Strategy, and Design (ASD) and located in the OneVA Enterprise Technical Architecture (ETA).

APPENDIX C: ACRONYMS

Acronym	Description
ABAC	Attribute Based Access Control
COTS	Commercial Off-the-shelf
GOTS	Government Off-the-shelf
FHIR	Fast Health Interoperability Resources
HATEOAS	Hypermedia as the Engine of Application State
HL7	Health Level Seven
HTTP	Hypertext Transport Protocol
JSON	JavaScript Object Notation
MUMPS	Massachusetts General Hospital Utility Multi-Programming System
NPE	Non-person Entity
OSEHRA	Open Source Electronic Health Record Agent
PE	Person Entity
PKI	Public Key Infrastructure
RBAC	Role Based Access Control
REST	Representational State Transfer
SAML	Secure Assertion Markup Language
SNOMED	Systemized Nomenclature of Medicine
SOAP	Simple Object Access Protocol
TLS	Transport Layer Security
VistA	Veterans Information Systems and Technology Architecture
XACML	Extensible Access Control Markup Language
XML	Extensible Markup Language