COLLA 2018

Validation of a New Component into an Existing Distributed System

Introduction

George Blankenship

24 June 2018

16:15 – 18:45

George Blankenship

- Academics
  - BS Georgetown university (mathematics)
  - MS Marymount (computer science)
  - DSC work at GWU (computer science)
- 40+ years in computer industry
  - Protocol design
  - TCP/IP implementation manager
  - Distributed system project manager
  - Systems programmer and consultant

Outline of Tutorial

- Introduction
- Collaboration
- Mature Distributed System
- Addition of a New System
- System Development Life Cycle
- Validation of a New System Requirements
- Validation of a New System
- Validation Demonstration
- Lessons Learned

Objective of Tutorial

- Explore the integration of a new component into an existing mature processing environment
- The objective the new component is to introduce a new capability desired by the end users
- The approach would also be appropriate for an update to an existing component
- The approach assumption is that the existing componentry is stable and reliable

Outline of the Tutorial

- Collaboration
- A complex mature distributed system
- A new capability for the system
- SDLC overview
- Requirements validation
- Target system validation
- Validation software
- Lessons learned

Rationale for the Discussion

- My experience has been focused on projected performance of a large distributed system
  - Apollo lunar landing program
  - Skylab program
  - Weather system upgrade program
  - Encrypted messaging
  - ARPANET migration to the Internet (informal organization)
  - GOSIP (national validation laboratories)
- Common thread is validation is based upon proper transmission
- Is this approach proper?

What is Collaboration?

- Definition
- Major components
  - Messages
  - Engines
- Human collaboration
- System Collaboration

Definition of Collaboration

- Cooperative environment where two (or more) entities work to a common goal
- The key item for a successful collaboration is a clear definition of the environment
  - The objective
  - The work items
  - The exchange of work items

Major Elements of Collaboration

- The messages are the encapsulation objects for the movement of the work items
- The actual path used by the entities is shared by each entity, but is independent of the collaboration objective
  - Voice or an electronic transmission
  - Common encoding for all entities
  - Common approach for message movement

Collaboration Engine Components

- The engines are the entities that consume the work items
- The message encoding used is shared by all entities, but is independent of the collaboration objective
  - Message items follow a standard encoding
  - Message items must have a clear definition
  - Message item must have a clear context

Human Collaboration

- The message engine in a human collaboration is the human being
- The message encoding is based upon language (vocabulary and grammar)
- The human being recognizes the message content
  - Information for a decision (future or immediate)
  - Request for information
- Example is text message conversation

System Collaboration

- The message engine in the collaboration of systems is an application program
- The message encoding is based upon a recognized standard (syntax and semantics)
- The application recognizes the message content
  - Write request to a database
  - Read request (simple read or complex computation)
- Example is an eCommerce interaction

A Mature Distributed System

- Department of Veterans Affairs
- Second largest agency of the US federal government
- Objective: to fulfill President Lincoln's promise "*To care for him who shall have borne the battle, and for his widow, and his orphan*" by serving and honoring the men and women who are America's veterans.

Health System

- **The Veterans Health Administration (VHA) is the largest integrated health care system in the United States**
- **More than 1,240 sites of care**
  - **170 medical centers,**
  - **1061 ambulatory care and community-based outpatient clinics**
- **More than 9 million people receive care (2018)**
- **Components of Interest**
  - **VistA (medical center)**
  - **MPI/MVI (patient identify)**
  - **HDR (clinical information**
  - **VSSC (outcomes analysis)**

VistA

- Veterans Information Systems and Technology Architecture (VISTA)
  - nationwide information system
  - Electronic Health Record (EHR)
  - developed by the U.S. Department of Veterans Affairs
- EHR system for a hospital system and its dependent work locations
- The hospital VistA complex must be able to support patient care when isolated

MPI/MVI

- Master Patient Index
- (additionally Master Veteran Index)
- Authoritative location of patient identity and selected other metadata information
- System register with the MPI for patient metadata information updates

HDR

- Health Data Repository
- Record of Veterans clinical data
- Display clinical data from VistA systems
- Prescriptions
- Vital signs

VSSC

- VHA Support Service Center Capital Assets
- Data warehouse
- Primary function of clinical interest is outcomes analysis

Objective of the New System

- Home Telehealth Program
- Support a large number of patients
- Remote monitoring of patient
- Care directed by Disease Management Protocol (DMP) with vital sign capture and video sent to a triage system
- Clinician uses collected data to closely monitor patient without bringing them to the hospital
- Benefits
  - Better level health with fewer emergencies
  - Happier patients
  - Lower costs

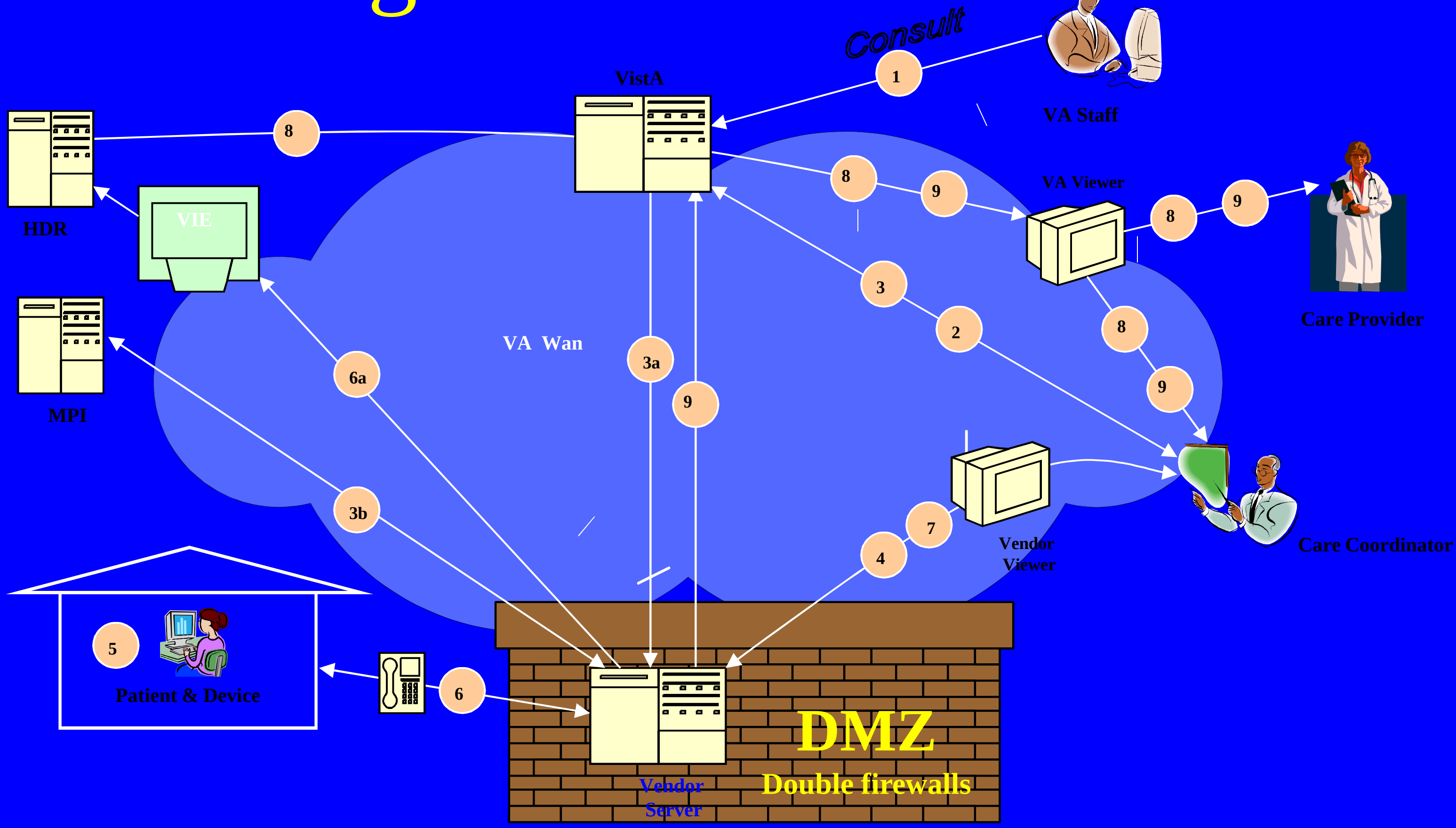
Description of the System

- Medical collection devices supplied to patient
  - Home device
  - Interactive Voice Response (IVR)
  - Independent mobile system (tablet, smart phone, …)
  - Internet connected
- Triage system
  - Receives data from collection devices
  - Performs a required care analysis based upon DMP
  - Clinical desktop supplies a ranking of care importance
- System creates a data island
- System is not accessible by CPRS (VistA GUI)

Integration Objective

- Eliminate the data island through integration with VA systems
- Patient identity is slaved to the MPI
- Patient data is placed in VistA and HDR databases using VA format definition
- CPRS user is able to access patient data

VistA Integrated Clinical View



1. VA staff requests patient be considered for enrollment by sending VistA consult.
2. Care Coordinator completes the VistA consult action:
3. Care Coordinator initiates registration of patient from Home Telehealth service using VistA
   - a) VistA sends HL7 sign-up message with patient identification to vendor system.
   - b) Vendor subscribes for MPI updates.
4. Care Coordinator links device with patient record and arranges to have device installed in the patient's home.
5. The patient uses the in-home device to capture vital signs and respond to any questions
6. Device exchanges information with vendor server, normally once per day
   - a) Vendor sends measurement data to the Health Data Repository (HDR) via the Interface Engine using HL7.
7. Using Vendor Viewer, Care Coordinator logs into vendor system to review patient information.
8. Care Provider and Care Coordinator can review Home Telehealth and VistA information in VA Viewer (CPRS and VistAWeb).
9. Vendor server sends draft Monthly Progress Notes to facility VistA server.

System Development Life Cycle

- Birth and Death of a System (SDLC Phases)
- (Phase I) Requirement gathering and analysis
- (Phase II) Design
- (Phase III) Implementation or coding
- (Phase IV) Testing
- (Phase V) Deployment
- (Phase VI) Maintenance

Integration Activity

- Basic objective is that the new system has “all” the aspects of the existing systems
- Phase I (performed before procurement)
  - Create document defining new system operational objectives
  - Collect documents defining existing environment
- Phase II (performed before procurement)
  - Generate requirements for new system
- Phase IV (performed as part of the RFU activity)
  - Validate new system performs as expected prior to operational deployment
- Phase V (performed as part of the RFU activity)
  - Validate new system performs as expected in operational test lab
- Phase VI
  - Monitor new system performance

Items of Concern

- Basic objective is that the new system has “all” the aspects of the existing systems
- (Phase I)
  - Documentation is stale as soon as it is published
- (Phase IV)
  - Is there an existing lab?
  - Is the existing lab able to support the required testing?
- (Phase V)
  - Is there a pre-deployment lab with all existing systems?
- (Phase VI)
  - Is there group responsible for system monitoring?
  - Is the monitoring adequate?

Approach to the Concerns

- Basic objective is that the new system has “all” the aspects of the existing systems
- (Phase I)
  - Build a validation lab to validate collected documents
- (Phase IV)
  - Build a validation lab to confirm compliance
- (Phase V)
  - Build a pre-deployment lab with existing application and representative database
- (Phase VI)
  - Build a monitor to evaluate the operation of the new system

New System Requirements

- Collect the set of documents defining current environment
  - The document set is drawn from each system in the existing environment
  - The documents is relevant within the environment, but may be of little use to external systems
- Draft of requirements document for new system based upon the collected documents
  - The documents must be relevant to an external perspective
- Validation of the requirements document

Reference Engine SDLC

- (Phase I
  - Preconditions are notional concepts for an actual requirements document
- (Phase II)
  - Requirements document for reference engine
  - Validation plan for use of reference engine
- (Phase V)
  - Validation of proper response by reference engine
- (Phase VI)
  - Update of new system requirements document
  - Return to Phase II, if missing requirements are discovered

Validation Engine SDLC

- (Phase I)
  - Requirements document for new system are final
  - Reference engine is the basis for the validation engine
- (Phase II)
  - Design document for modifications to reference engine
  - Design of test scripts for new system validation
- (Phase V)
  - Validation of proper responses by reference engine and the validation engine
- (Phase VI)
  - Update due to VA system (existing environment) change

Validation Tests

- The validation uses a black box approach
  - Testing starts with a precondition setup in the validation engine and the SUT
  - No modification is required in the SUT
  - The SUT operates normally as if it were in a production environment
  - The validation engine uses a finite state machine (FSM) to identify that the received messages are proper
- A validation test is a set of message exchanges
  - One system will start a processing sequence (message sequence)
  - The validation engine uses a script to direct the responses to SUT messages

Validation of a New System

- Preconditions
  - Validation of a New System
  - Reference engine has validated the new system requirements document
  - The validation test plan is complete
  - Validation lab exists
- New system passed internal SDLC
  - Any problem found during a new system validation returns the system to a **not RFU state**
  - The new system must be **RFU** (by their measure) before validation

Validation Laboratory

- Internet based laboratory
  - All test sets defined
  - Reference engine is the basis for the validation engine
- (Phase II)
  - Design document for modifications to reference engine
  - Design of test scripts for new system validation
- (Phase V)
  - Validation of proper responses by reference engine and the validation engine
- (Phase VI)
  - Update due to VA system (existing environment) change

Validation Environment

- New System owner laboratory
  - System Under Test (SUT) resides in the system owner lab
  - The SUT administrator is single point of contact
- Validation laboratory
  - Validation engine emulating existing collaboration environment
  - Test conductor administers the testing
  - The validation engine generates testing results in an email to the SUT administrator

Validation Demonstration

- Testing laboratory used for the demonstration is the one used for the validation of new systems for the VA EHR environment
- **BLANKENSHIP11**5 supplies the **SUT**
- **EITL** supplies the **validation engine**
- I am the test conductor, there is no SUT administrator
  - A test will be started that does not require an SUT setup
  - The test administrator will start the test
  - No manual intervention is required
- Upon completion, the validation engine sends an email to the SUT administrator with a summary report

Lessons Learned

- Probability of an exceptional condition is non-zero.
  - A portion are the result of code defects
  - A portion are the result of defective intermediate entities
  - A portion are the result of environmental issues
  - The processing of the condition could be very interesting
- A collaborative processing distributed system must have a minimal amount of manual intervention due to delay and erroneous responses
- A distributed system must address the ability for disconnected operation
- A distributed system must address redundancy

The Take Away

- Very seldom does testing take exception conditions into account
- The norm is to expect that peer systems are reliable and operate error free
- Probability of an exceptional condition is non-zero.
  - A portion are the result of code defects
  - A portion are the result of defective intermediate entities
  - A portion are the result of environmental issues
  - The processing of the condition could be very interesting

---

- "***In general, an implementation must be conservative in its sending behavior, and liberal in its receiving behavior.***"
  - Jon Postel (one of the founders of the Internet)