Assessing Fault Management in Network Resource Intensive and Protocol Rich Environments

Presented by Tom Hempler
September 11, 2014

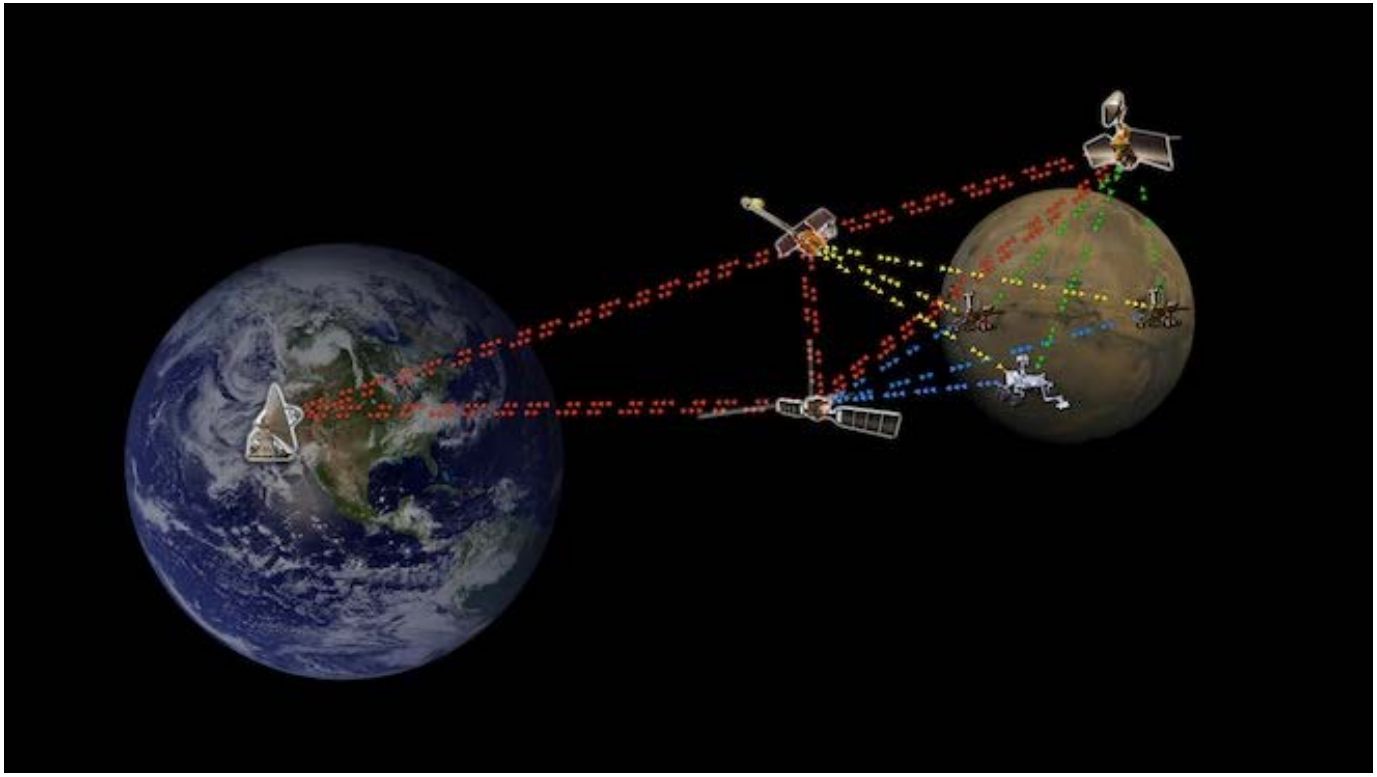


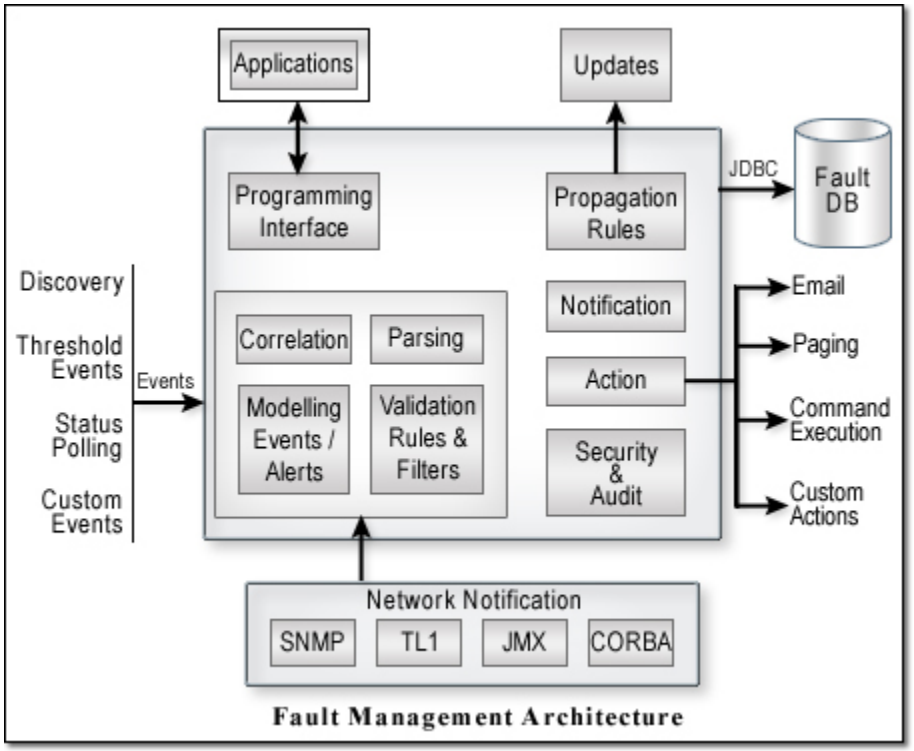
Image: NASA/JPL

Fault Management Verification

- Ensure network infrastructure functions correctly and supports current and future space network needs.
- Ensure architecture malfunctions and environmental conditions do not impact functional capabilities of the system or its reliability.
  - A satellite reporting lightning strikes should not mistake them for meteor showers
  - Communication dropouts due to solar storms and and communication delays due to spatial distance should not be mistaken for disconnects or access faults
- Predictable errant behavior is challenging
- Unknown, unexpected, or unpredictable events are more challenging
  - A parked motor vehicle should not start moving when the engine is started, but it did
  - Commanding satellite switchover from solar to battery power source prior to eclipse is predictable and should not be delayed or interrupted but it can be
- Fault management design must support reliable and durable network processing and communications

Network Architectures and Fault Management

- Typical fault management in network resource intensive and protocol rich environments



Network Resources and Protocol Reliability

- Fault management analysis ensures:
  - Architecture and services reliability is met in requirements, design, and implementation
  - SOA messaging reliability and durability is maximized for high availability
- Assess compliance of infrastructure with protocol standards
  - Measure correct, consistent, predictable network behavior
  - Measure degree of compatibility among interface technologies (precision, range deltas)
  - Enumerate known vulnerabilities of protocols being implemented
- Assess fault management quality with reliability measures in four dimensions, in proven practices, and statistically significant values
  - Measure 1) Availability, 2) Reliability, 3) Safety and 4) Security
  - On a very large architecture and infrastructure measurable quality attributes include fault management coverage, message durability and reliability, resources and services availability, and security accountability
- Evaluate services reliability – examine network reliability and messaging durability

http://blogs.msdn.com/b/nickmalik/archive/2007/05/27/reliability-in-soa-is- huge.aspx

Network Resources and Protocol Dependability

- System Dependability Is Strengthened Through
  - Error handling that is detectable and recoverable at all Levels of the network topology, protocols, applications and resources
  - Fault Management that is measured at all Levels of the Network Topology, Protocols, and Applications through exception handling frameworks among the technologies
  - Recovery Management is only as effective as error and fault detection
- Factors stressing dependability are:
  - Network latencies
  - Data bandwidth
  - Communication link distances
  - Network security threats
- Verification processes quantify error handling and fault management coverage and capability
- Identify risk

Measuring Network Management Functions – FCAPS Model

- Fault, Configuration, Accounting, Performance, and Security (FCAPS) model is the industry standard for network management
  - Managing FCAPS are key management features in current technology for meeting goals
  - Fault management enables fault detection, correction, and network recovery
  - Network protocol and resource exception handling facilitates fault management
  - Tradeoffs in performance and FCAPS implementation drive risk
- Fault management throughout system development to deployment
  - Requirements – specify functions for operations, interoperability, and failure mitigation
  - Design – built-In alarming, isolation, diagnostics, error handling, alerting, and error logging
  - Implementation – build-in events, operator and message validation, status, and exception handling
  - Integration and Test – unit test for compile time errors and subsystem tests for exceptions
- Fault management during operations
  - Collecting and analyzing logs, correlating events, identification and authentication failures
  - Protect safety critical software from emergent events
  - Data storage, archiving and playback to facilitate forensics
  - Protect or backup control layer and data layer, and provide multiple path decision making processes

Protocols and Frameworks

- SNMP
- ICMP
- Java Exception Handling Framework
- Availability Management Framework SAI-AIS-AMF-B.04.01
  - http://www.saforum.org/hoa/assn16627/images/sai-ais-amf-b.04.01.al.pdf
- JAVA Community Process JSR-000319 Availability Management for Java
  - http://download.oracle.com/otndocs/jcp/availability_management-1.0-fr-oth-JSpec/

Protocols – Lightweight Directory Access Protocol (LDAP) Error Codes

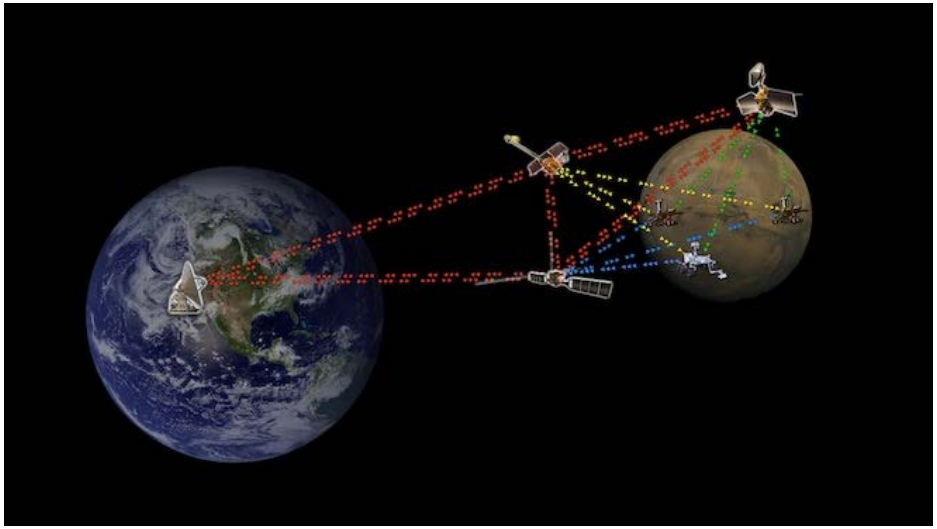
- LDAP error codes facilitate exception handling for directory services
  - Provides a fault management capability in network infrastructure
  - LDAP defines a set of status codes that are returned with LDAP responses sent by the LDAP server (see RFC 2251). LDAP error codes map to JNDI Exceptions.
  - In the JNDI, error conditions are indicated as checked exceptions that are subclasses of NamingException.
  - LDAP service provider translates the LDAP status code it receives from the LDAP server to the appropriate subclass of NamingException.
  - LDAP status codes are useful for troubleshooting and managing network directory services; and detecting operations, protocol, authentication errors to name a few.
- IV&V analyzes fault management quality and coverage of LDAP authentication and directory services

Protocols – Simple Object Access Protocol (SOAP) Faults

- SOAP Faults facilitate exception handling for web service messaging
  - Provides a fault management capability over network infrastructure (i.e. distributed databases)
  - SOAP Fault Element <wsdl:fault> in SOAP message contains:
    - <Code> <Reason> <Node> <Role> <Detail>
  - Analogous to an application exception, generated by message receivers to report business logic errors or unexpected conditions.
  - The faults returned to the sender only if request/response messaging is in use.
  - If a Web service operation is configured as one-way, the SOAP fault is not returned to the sender, but stored for further processing.
- SOAP Fault and Java Exception Mapping
  - Modeled – Maps to an exception that is thrown explicitly from the business logic of the Java code and mapped to wsdl:fault definitions in the WSDL file, when the Web service is deployed. In this case, the SOAP faults are predefined.
  - Unmodeled – Maps to an exception (for example, java.lang.RuntimeException) that is generated at run-time when no business logic fault is defined in the WSDL. In this case, Java exceptions are represented as generic SOAP fault exceptions
- IV&V analyzes fault management quality and coverage of SOAP messaging

Protocols – IP and Delay Tolerant Network / Disruption Tolerant Network

- Variable delay and disruption due to planetary distance and rotation
- TCP/IP protocols designed for limited router memory.
- Packets are discarded due to memory constraints
- Enter Bundle protocols such as CCSDS, DTN – using Packet Switching Store and Forward techniques
- Bundle Security Protocol – RFC 6257



- In 1973 Bob Kahn and Vint Cerf developed the internet protocol suite (TCP/IP)
- In 1997 Vint Cerf began working on interplanetary communications
- Cerf working with NASA & JPL helped develop a new set of protocols that can withstand the unique space environment for an interplanetary internet
- Planetary Rotation, Time Synchronization, Domain Naming and Addressing, Routing and Security are among the challenges for IP
- Delay Tolerant Network (DTN) under active development by DTNRG
- DTN experiments on ISS and Mars
- BP allows for disconnections, glitches and other problems commonly found in deep space

http://www.wired.com/2013/05/vint-cerf-interplanetary-internet/

White Papers

- ManageEngine
  - COTS product that provides FCAPS for virtual environments
  - http://www.manageengine.com/products/white-papers.html

TASC

INSIGHT APPLIED™