National Aeronautics and Space Administration

# Goddard Mission Services Evolution Center (GMSEC)

# Overview

***May 2011***


Ryan Detter

NASA Goddard Space Flight Center

Software Engineering Division

GMSEC-Support@lists.nasa.gov

301-286-7325

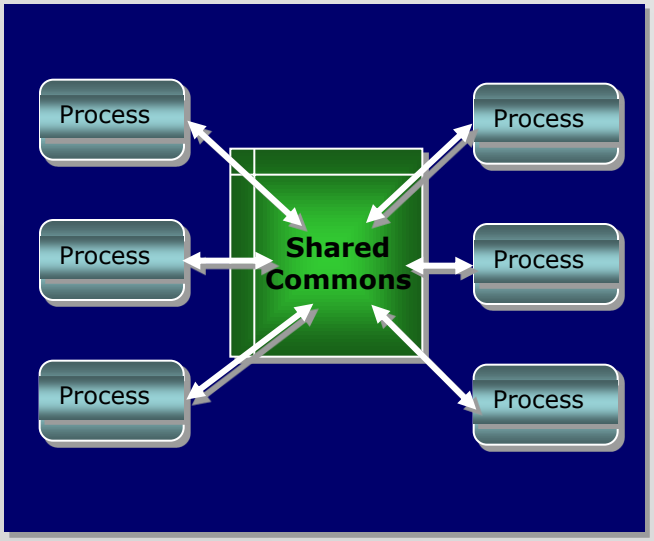

# GMSEC Background and Introduction

GMSEC was established in 2001 to coordinate ground and flight data systems development and services at GSFC

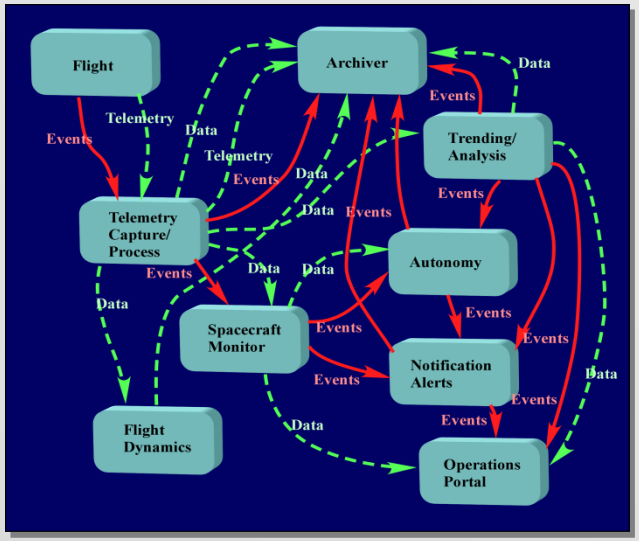
- Goals
  - Simplify development, integration and testing
  - Facilitate technology infusion over time
  - Support evolving development and operational concepts
  - Allow for mix of heritage, COTS and new components while avoiding vendor lock-in
- Concepts
  - Standardize interfaces – not components
  - Provide a middleware infrastructure
  - Allow users to choose – GMSEC doesn't decide which components are best or dictate which components a mission must use. It's the mission/user's choice!

# History of Software System Architectures

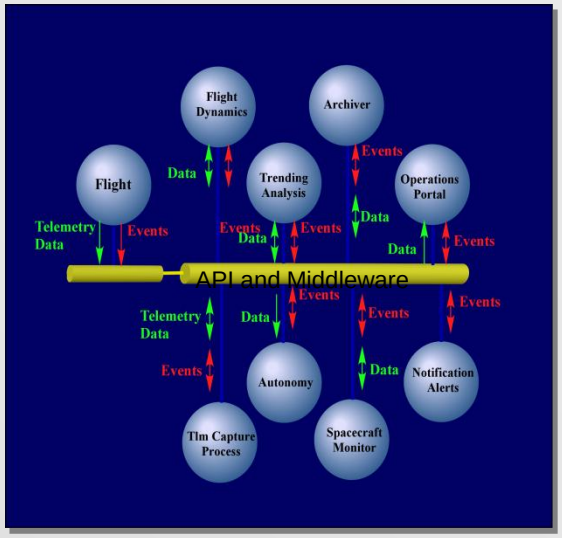
Shared Commons
(1970's -1980's)




Traditional Design
Socket Connections

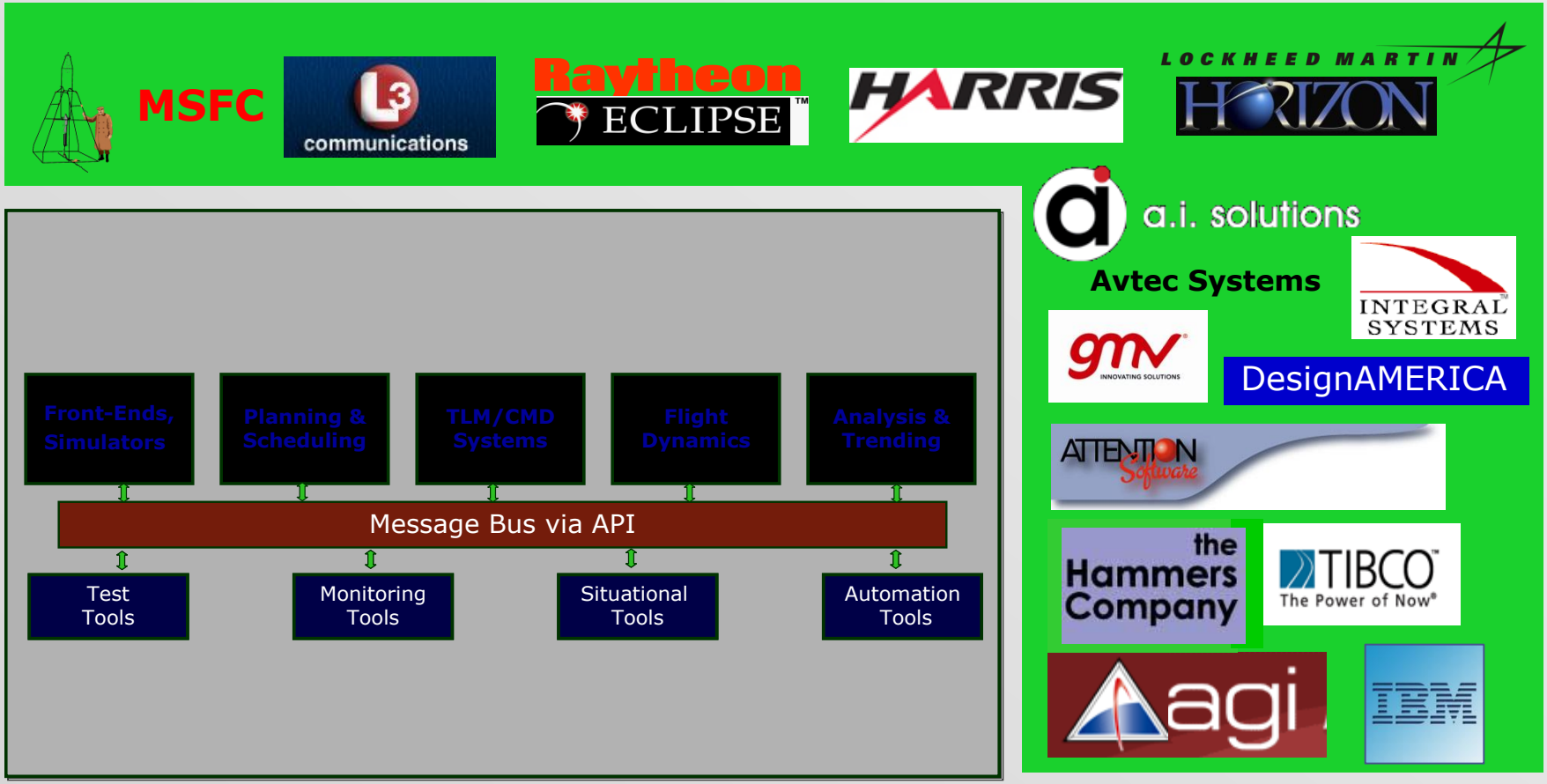



Current Advanced Designs
Middleware Connections



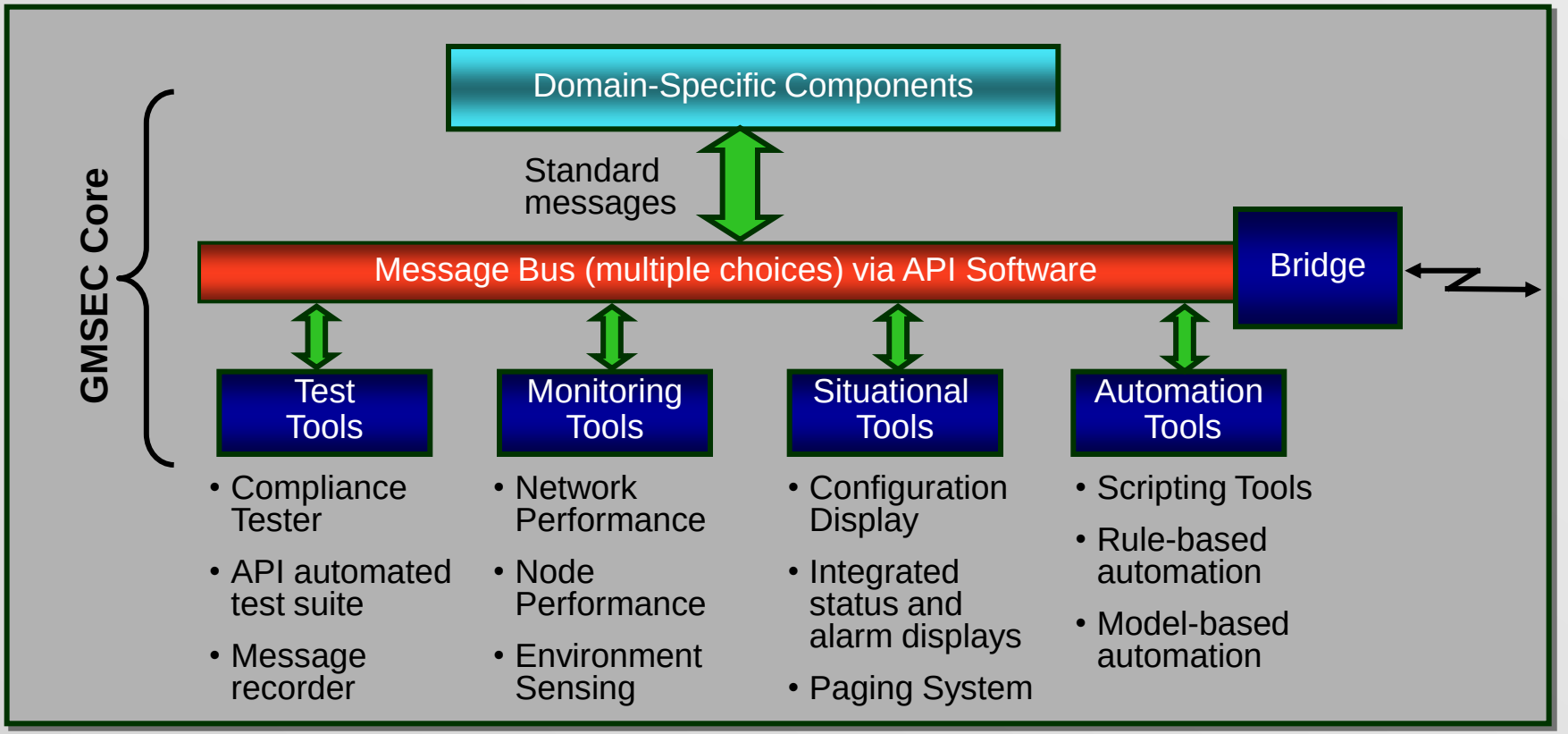

***GMSEC software applies modern development approaches to ground system applications.***

# GMSEC and the Satellite Control Domain




***Users can choose the best products for their needs. Nearly all COTS command and control systems are now GMSEC compatible.***

# Core GMSEC System Supports any Domain




***GMSEC's common service tools bring immediate value to any system.***

# Application Programming Interface (API)


- Defines the functional interface between application components and the middleware. It normalized middleware behavior where possible and standardized message formats. Developers use the standard API to interface with the underlying middleware, and the standardized messages to interface with other components. API supports a number of programming languages, COTS and GOTS middleware products, and operating systems.

- **Programming languages**: C, C++, Java, and Perl
- **Middleware Products:** TIBCO SmartSockets, Apache ActiveMQ, IBM WebSphere MQ, GMSEC Message Bus
- **Operating Systems:** Microsoft Windows XP (32 & 64 bit), Microsoft Windows 2003 Server; Red Hat 3, 4 & 5 (32 bit); Solaris 10; Mac-Intel OSX; Red Hat 5 (64 bit)

# API (cont)

- **Features:**

- Provides a set of common message communication functions with both synchronous and asynchronous delivery mechanisms:
- Connection Services: connect, disconnect
- Message Services: create message, add field(s) to message, get next message, and destroy messages
- Message Interchange Services: publish/subscribe, request/reply
- Each application uses the connect service to establish communications with the middleware, and uses the publish/subscribe and/or request/reply services for passing messages. Applications subscribe only to those desired messages.

# GMSEC Components

# System Agent (SA)

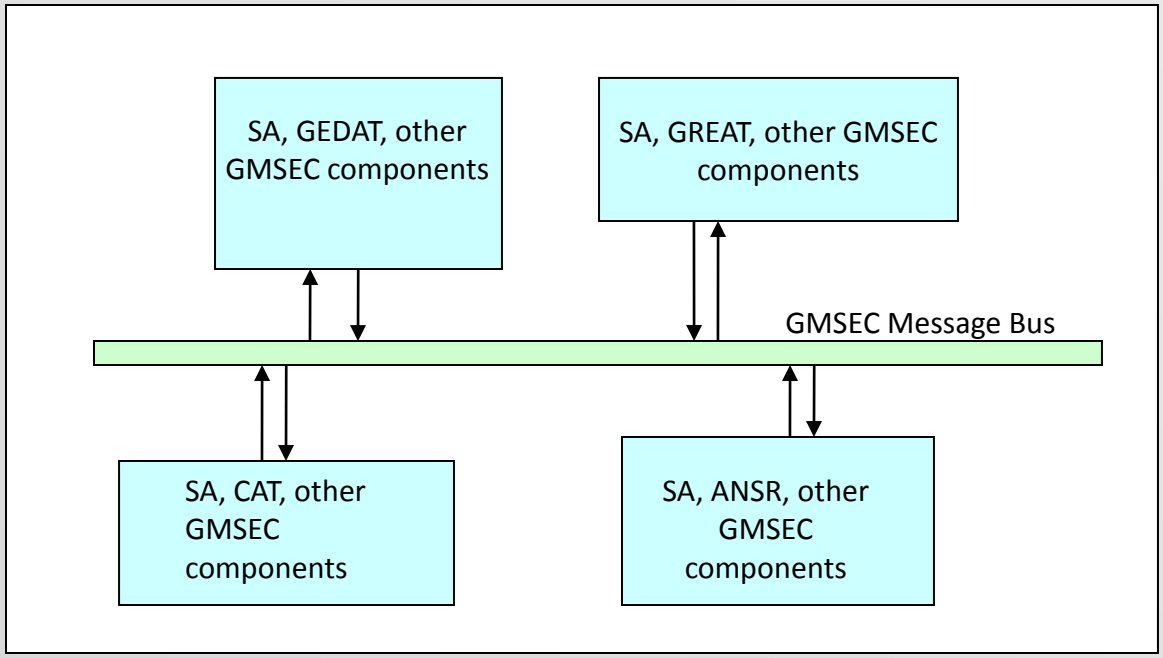
**Overview**

The GMSEC System Agent (SA) is a GMSEC-compliant software component that provides health information about the computer hosting the agent to other GMSEC components utilizing a middleware-based architecture. Monitoring more than one workstation can quickly turn into a tedious and expensive task. Having an agent that reports a machine's health status using the middleware paradigm can support a more generic, automated, and centralized monitoring tool for the entire GMSEC system.

**GMSEC Features**

1. Messages can be displayed using GMSEC Environment Diagnostic Tool (GEDAT) allowing Flight Operations Teams (FOT)s to monitor the health of the entire GMSEC system in one centralized location.
2. Resource messages contain information such as: Percentage of CPU utilized, Amount of total physical memory, Percentage of each disk space among others.
3. Criteria Action Table (CAT) rules can be used to send commands to SA for error detection and correction. Ex. If the CPU usage is too high, CAT will send a Directive Request to the SA, and the agent will take a corrective action such as rebooting the machine.

# System Agent




GMSEC system using SystemAgent

# System Agent Demo

# Criteria Action Tool (CAT)

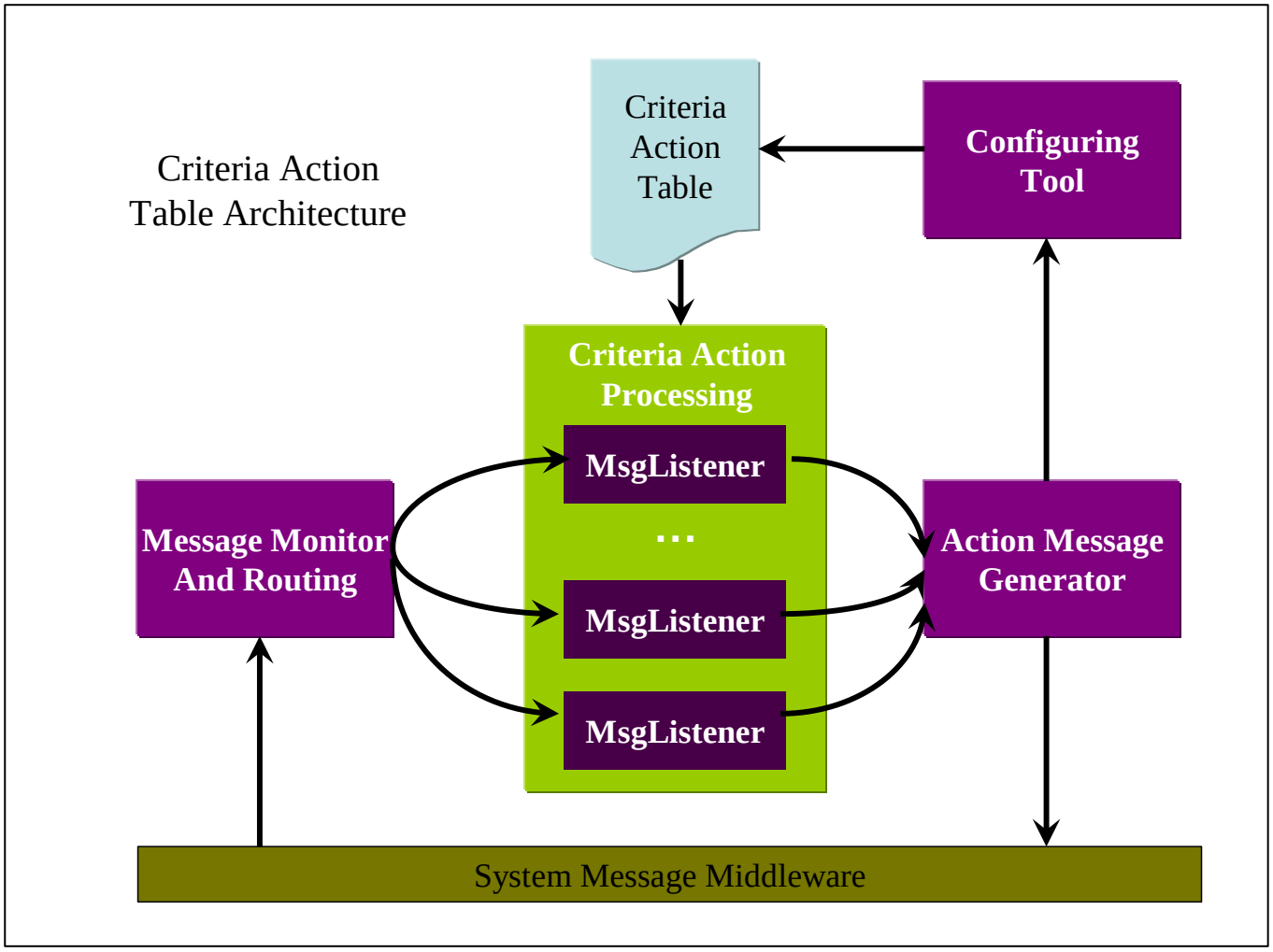
**Overview**

The Criteria Action Table (CAT) is an autonomic computing tool for spacecraft ground systems under the GMSEC architecture. CAT captures data from its managed elements and environment, performs the data analyses to generate the actionable data, and makes decisions based on the combination of the actionable data and the rules or policies from the management. This results in actions being sent as either GMSEC standard Directive messages or event Log messages to the relevant component.

**GMSEC Features**

- Automates detection of anomalous conditions
- Provides more automation and autonomy to mission operations
- Assists in efficient ground system operations
- Non-Intrusive to Current Missions (monitor only mode)
- Simple to complex monitoring criteria are user configurable
- GUI rule editor to allow easy creation of rules and monitors

# CAT Diagram


Criteria Action Table Architecture
Criteria Action Table
Configuring Tool
Criteria Action Processing
MsgListener
...
MsgListener
MsgListener
Message Monitor And Routing
Action Message Generator
System Message Middleware

# CAT/Editor Demo

# GMSEC Environmental Diagnostics Analysis Tool (GEDAT)

**Overview**

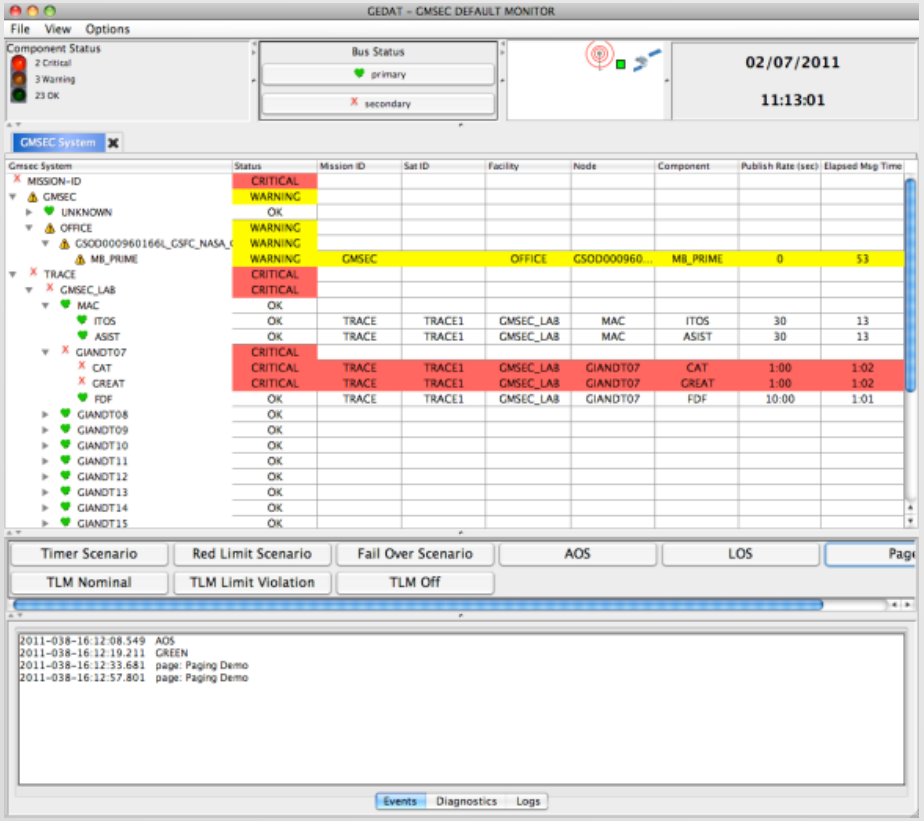
The GMSEC Environment Diagnostic Tool (GEDAT) provides a visual representation of the GMSEC environment. The display shows numerous network components performing message-based publish/subscribe communications via one or more GMSEC message buses. The display is intended to be a common software application capable of displaying data system activity at any level within the network.

**Key Features**

- Identifies components and tracks status by monitoring heartbeat messages
- Provides drill down access to detailed component information
- Display can be initialized with an expected environment
- Collects and plots node resource information (CPU, memory, network, disk utilization)
- Brings attention (audible/visual) to key events such as spacecraft passes, telemetry downlinks, alert notifications
- Provides user configurable buttons to publish messages

# GEDAT (cont)


GEDAT - GMSEC MONITOR
File View Options
Component Status
0 Critical
0 Warning
2 OK
Bus Status
primary
secondary
05/19/2011
19:39:58
GMSEC System
GS580W-GITEST03
CPU Utilization
Memory Utilization
Disk Utilization
Percent
Percent Used
Time
CPU-1
MEM-1
C:\
AOS
LOS
Page
TLM Nominal
Timer Scenario
Red Limit Scenario
Fail Over Scenario
Events
Diagnostics
Logs


GEDAT - GMSEC DEFAULT MONITOR
File View Options
Component Status
2 Critical
3 Warning
23 OK
Bus Status
primary
secondary
02/07/2011
11:13:01
GMSEC System
Timer Scenario
Red Limit Scenario
Fail Over Scenario
AOS
LOS
TLM Nominal
TLM Limit Violation
TLM Off
Events
Diagnostics
Logs

# GEDAT Demo

# GMSEC Reusable Events Analysis Toolkit (GREAT)

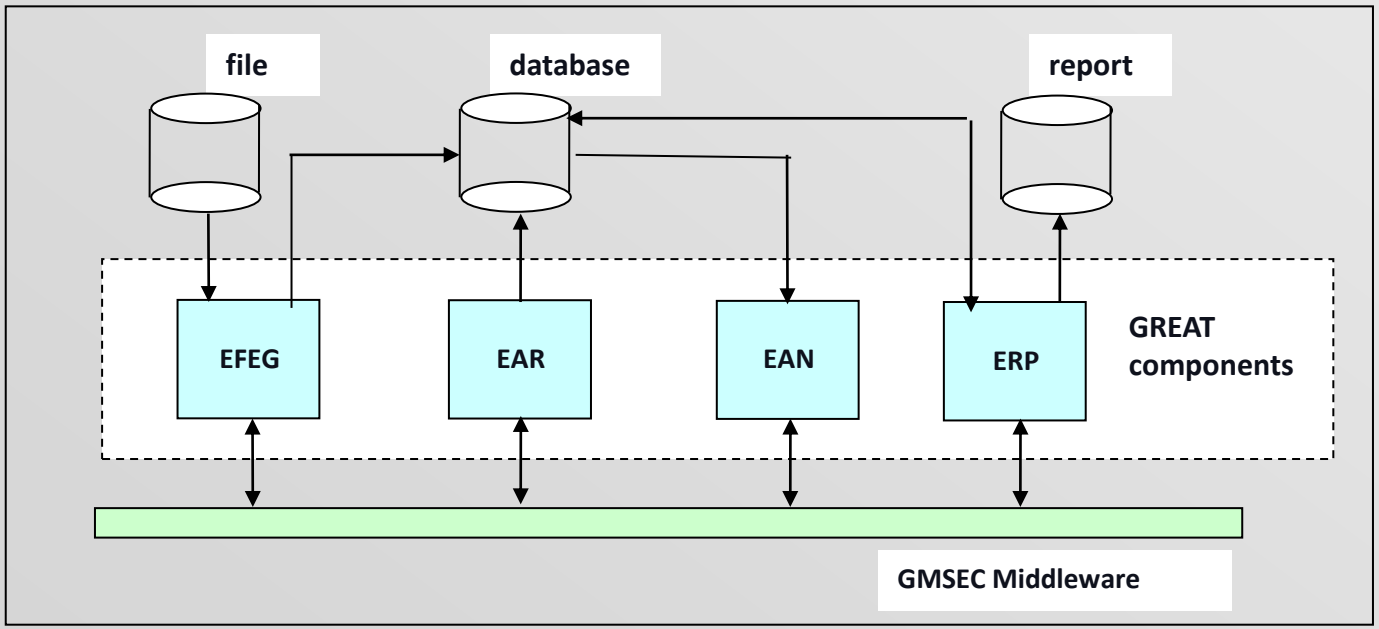
- GREAT is a collection of GMSEC compliant components for analyzing messages in a GMSEC architecture system. It is consists of 4 tools:

- **Event Analyzer (EAN)** receives message over the middleware according to the subscription(s).
  - Remote Access Event Analyzer (Historical Retrieval) – remotely query the Event Archive database for archived GMSEC messages via the Event Analyzer Remote Query website. It is configurable via the XML configuration file. Also include the option of saving the result to html file.

- **Event Archive (EAR)** receives messages over the middleware according to the subscription(s) and stores them in the database, which provides a platform for event retrieval, event analysis, and report generation. It also accepts directives to insert a record into the database using information contained in a message field.

# GREAT (cont)

- **Event Report (ERP)** provides the capability to create and/or edit customized report template(s) and accepts directives over the middleware to generate reports. The process of report generation involves data retrieval from the database and populates the database field in the template.
- **External File Event Generator (EFEG)** ingests an external file and based on its contents either replays those messages or inserts messages into database for future retrieval.



# GREAT Demo

# Room Alert Adapter (RAA)

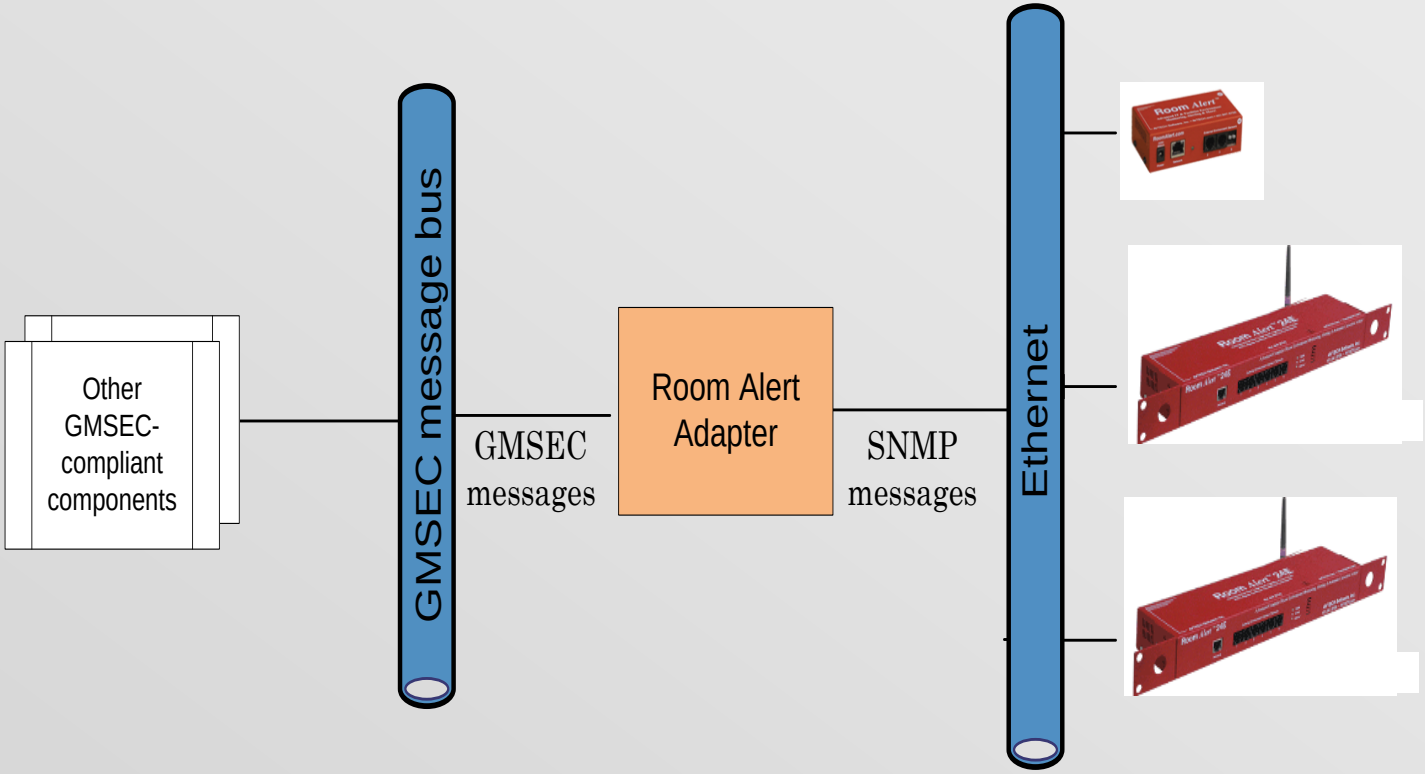
**Overview**

The GMSEC Room Alert Adapter is a GMSEC-compliant software component that provides an interface between AVTECH Room Alert™ environmental devices and the GMSEC message bus. The Room Alert™ device supports various sensors including: temperature, humidity, power, flood, etc. The devices enables 24/7 remote situational awareness of environment conditions, as well as detects and reports anomalous conditions with immediate alert notifications (email, email-to-SMS, SNMP, web page).

**GMSEC Features**

1. Parameters from each Room Alert device can be shown on a local display (GEDAT, GPD).
2. Alerts and status can be archived & retrieved using a GMSEC-compliant event analysis toolkit (e.g. GREAT).
3. Automatic corrective action, such as failovers, can be defined and implemented with a configurable rule-based component (e.g. CAT).
4. Comprehensive paging and notification capabilities are available using the GMSEC ANSR or Attention Software.

# Room Alert Adapter




Room Alert Adapter Architecture Diagram

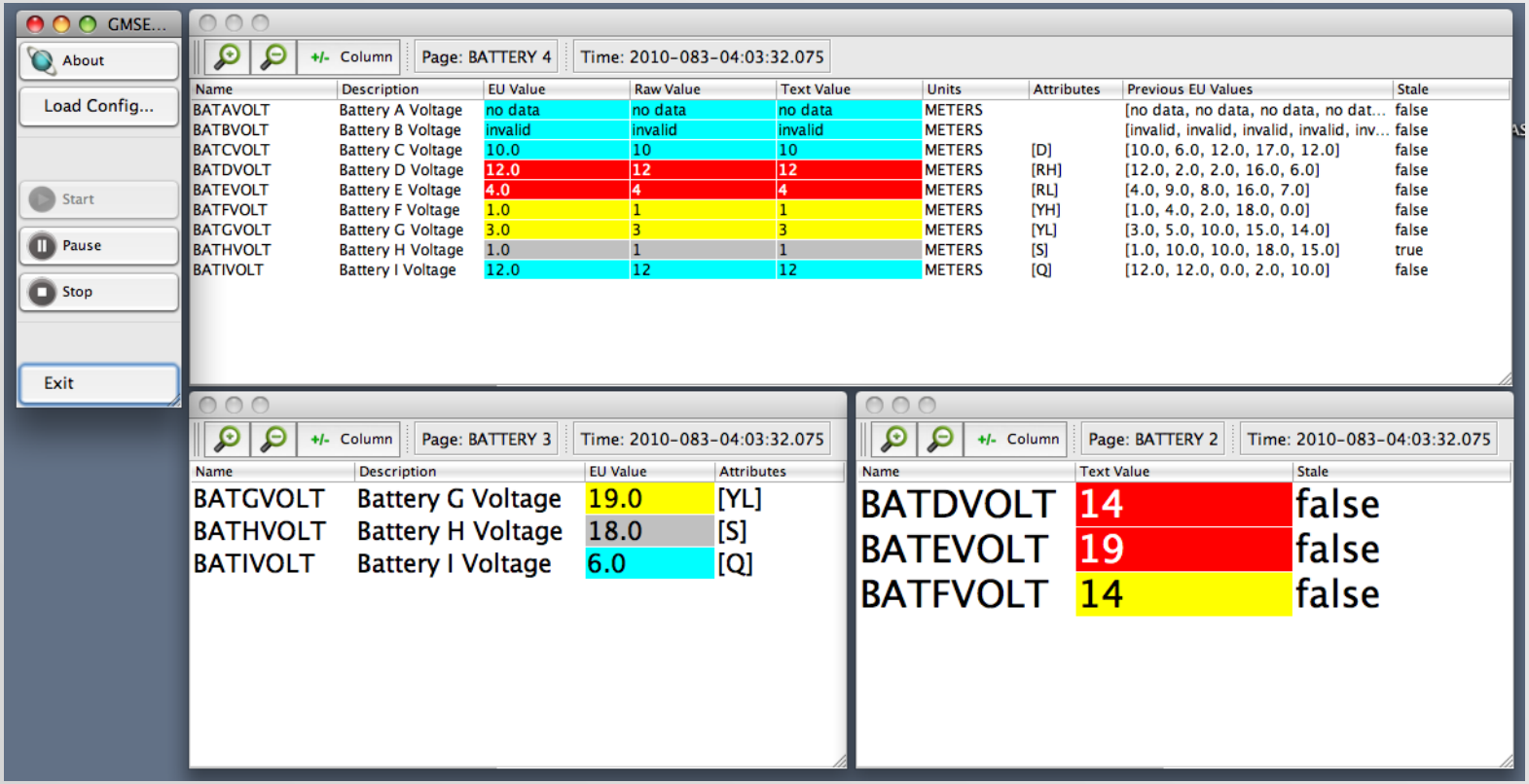
# GMSEC Parameter Display (GPD)

**Overview**

The GMSEC Parameter Display is a GMSEC-compliant software component that enables the users to quickly create and view a display page consisting of parameter and telemetry mnemonic values. The GPD operates within a GMSEC architecture-based system. As such, it utilizes standard GMSEC messages to request the user-defined parameters from various data providers. The parameters and their values received from the data providers will be displayed, as is, to the display pages.

In addition to displaying the telemetry values, the GPD also displays the attributes associated with the telemetry mnemonic values. Both text and color are used to show the mnemonic attributes and provide a visual delineation of their status. From the display, the user can quickly assess the value and state of the displayed parameters.

# GMSEC Parameter Display



# RAA→GPD Demo

# Alert System Notification Router (ANSR)

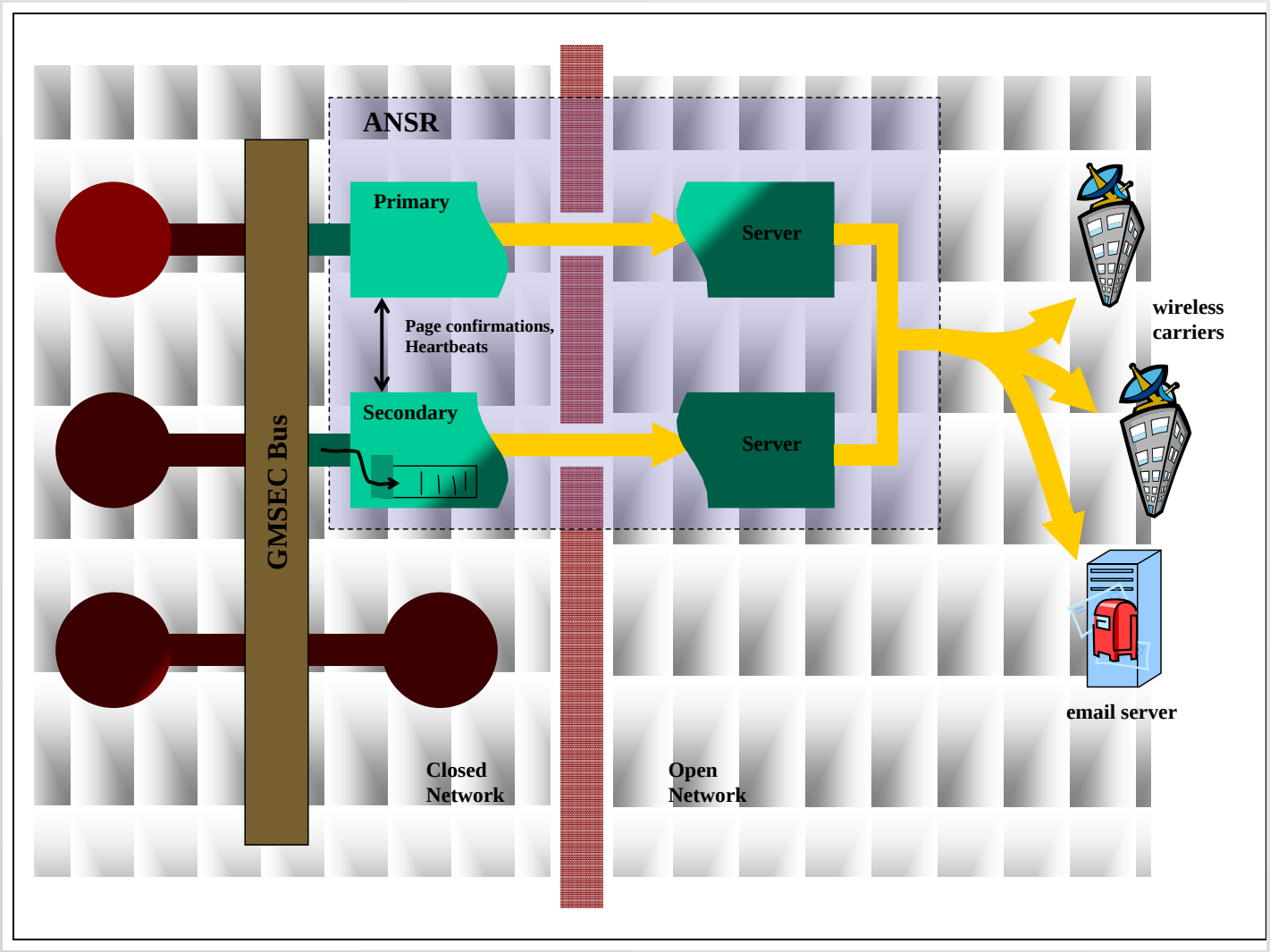

**Overview**

The GMSEC Alert Notification System Router (ANSR) is a GMSEC-compliant software component used in mission operations for operator notification and paging. Notifications are done through standard devices, such as pagers and emails. ANSR is commonly used in GMSEC supported ground systems as the notification gateway between an automated system and the operators.

**GMSEC Features**

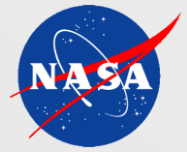
1. ANSR can be instructed to send pages by other GMSEC-compliant components, for example, using the Criteria Action Table (CAT)
2. ANSR broadcasts the status of pages to the entire GMSEC system, but does not require any component to listen or react to these messages. Other components, such as the GMSEC Reusable Event Analyzer Toolkit (GREAT), can collect and archive the messages.
3. ANSR can be configured to select which operators to be paged and e-mailed, as well as configuring if the messages need to be acknowledged or not, among other options.

# Alert System Notification Router (ANSR)



# ANSR Demo

# GMSEC Remote Application Service Provider (GRASP)



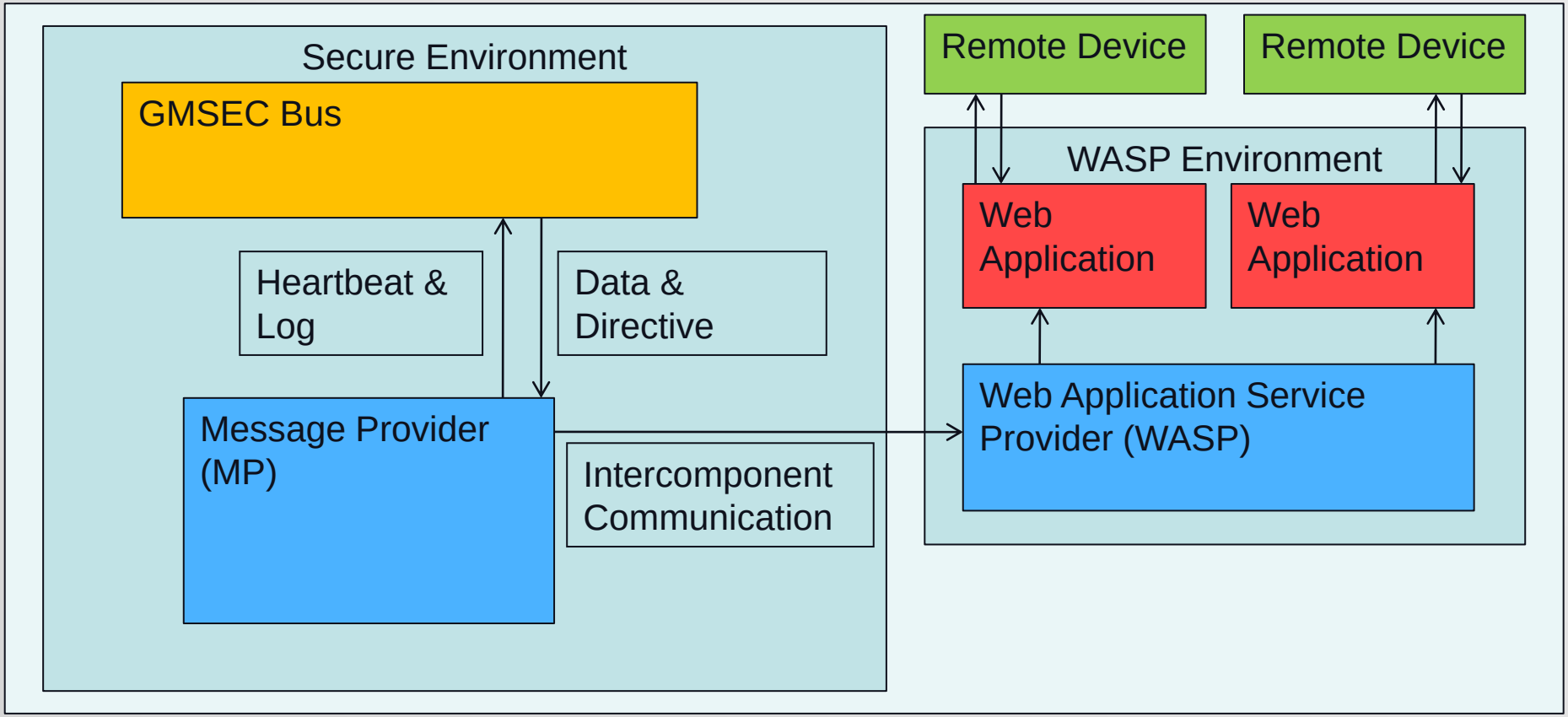

**Overview**

The GMSEC Remote Access Service Provider (GRASP) is a GMSEC-compliant software component that provides the capability for users to access selected GMSEC messages from a web server operating in an unsecure space without risk of contamination to the secure GMSEC network. It also provides access to selected functional capabilities via the GMSEC Directive Request message.

**GMSEC Features**

1. Provide a means for messages to be sent from a secure GMSEC network to a web server, ensuring that the messages are encrypted and signed, if GRASP is configured to encrypt and sign.
2. Provide the administrators of the secure GMSEC network the ability to control by subject which messages are pushed to the web server.
3. Provide assurance that no information from the web server shall be placed in the secure GMSEC network.
4. Provide a general-purpose capability for messages to be available to web applications. The messages must be available in such a way that they can be accessed by any Java based web application framework.

# GRASP Diagram



# GRASP Demo

# Red Limit Passover Scenario

Show Flash

# Component and Middleware Failover

# Final Notes

- GMSEC is not a radical technology
- GMSEC currently being used by many operational missions
- Many in-house and out-of-house components that are GMSEC compliant to meet mission requirements
- The value is that it addresses key issues for a certain class of users
  - Allows for use or mix of heritage or COTS software
  - Evolvable over time
  - Does not assume “one size fits all” will work
  - Does not enforce a specific operations concept
  - Simple enough for vendors to invest in
- Air Force, NRO, ORS are moving in this direction
- Appropriate architecture for more than just satellite control