

GMSEC

GSFC Mission Services Evolution Center



At A Glance

The GMSEC Information Message Bus provides mission enabling software communications services.

Features

- Application Programming Interface (API)
- Standardized messages
- Available in multiple languages, and swappable operating systems and middleware
- Easy implementation

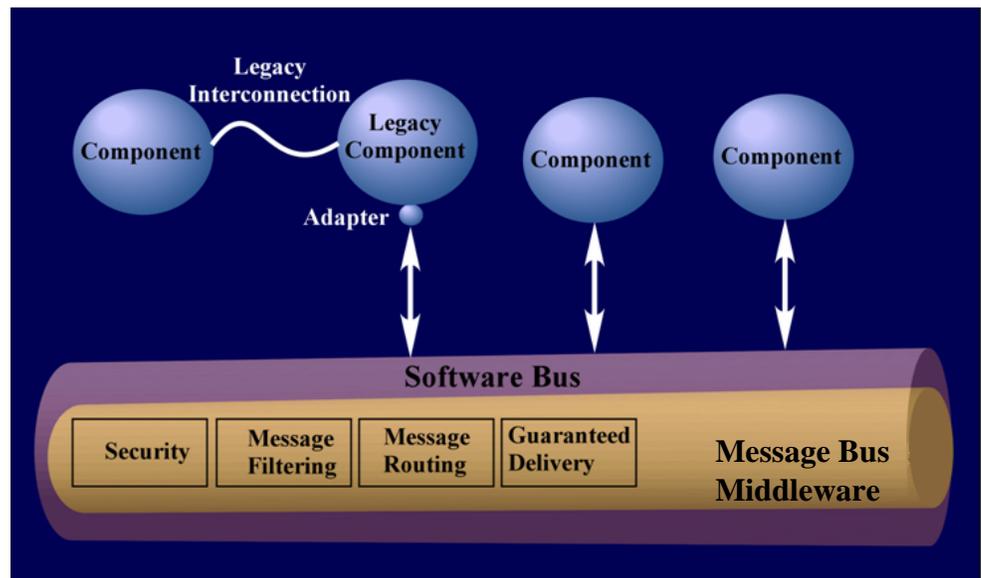
Benefits

- Greatly reduced & simplified system integration
- Supports evolving operations concepts such as automation
- Allows growth of current and future GSFC missions
- Achieve swappable, plug-and-play software components

NASA's GMSEC Information Message Bus

Summary

The GMSEC Message Bus is a middleware-based system architecture for NASA/GSFC missions using standardized messages, an applications programming interface (API), and commercial-off-the-shelf (COTS) and GOTS (government) components. It is a comprehensive flight and ground data systems framework that enables quick and easy system integration of functional components that are selected to meet the unique needs of a particular mission. Software components incorporate the GMSEC API and use standardized messages to interface to the Message Bus. The underlying middleware takes care of message routing and communications management thereby simplifying the components and freeing them of this responsibility and complexity.



Components Interface to the GMSEC Information Message Bus

Implementation

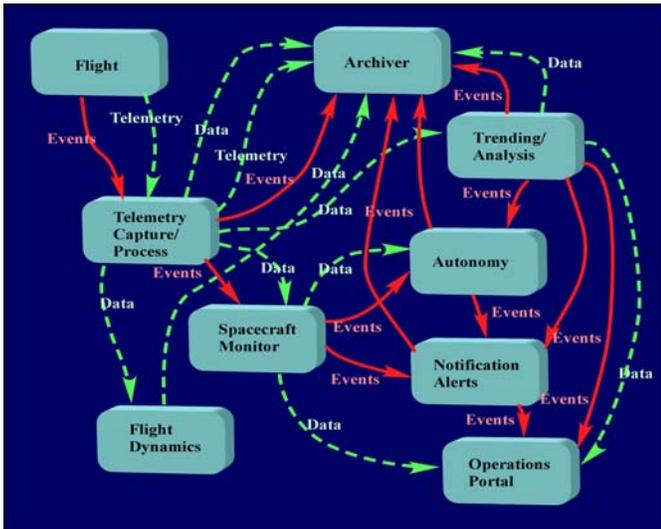
The Message Bus was designed to provide standardized interfaces (not components) using a middleware infrastructure. There were three facets to the design. The first was to provide proven, readily available middleware to relieve the applications of the communications responsibilities (and headaches). The middleware can range from minimal to one with a full capability set. Second, provide a standard common API to interface to all the middlewares, normalizing their behavior where practical. Third, was to standardized the data, status, and control messages passed between the components. By combining the standardized API, messages, and middleware components now interface to the Message Bus and not to each other. Thus, the number of connections are greatly reduced, and consequently, system integration. Additionally, the middleware and components themselves are easily swapped in and out of the system.

NASA GSFC Mission Services Evolution Center, Code 581
Greenbelt, Maryland 20771

<http://gmsec.gsfc.nasa.gov>

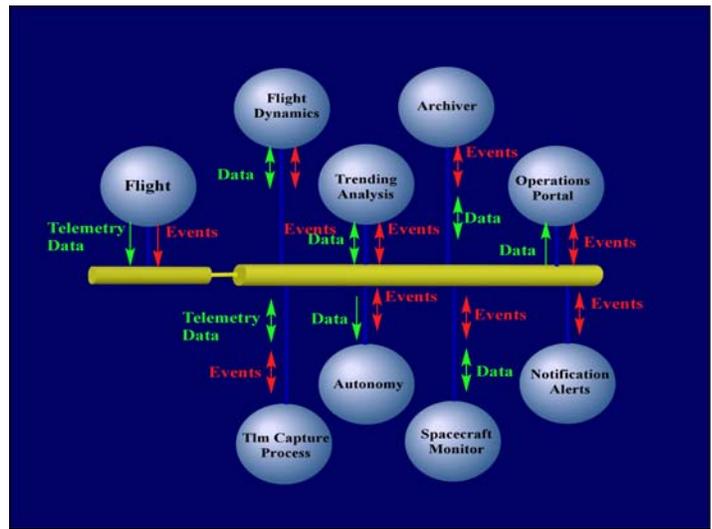
email: gmsec@nasa.gov

Traditional Design With Socket Connections



The diagram above shows the traditional approach of highly coupled component-to-component socket connections (spaghetti). The addition or removal of a component or interface will cause significant perturbations to the connections and data flow.

Message Bus Design With Middleware



The diagrams above and below show the components interfacing to the GMSEC Message Bus (not to each other) in a layered (lasagna) approach. The addition or removal of one component has little to no effect on the other components.

Enhanced Capabilities

System Wide Monitoring and Display

- By monitoring the status messages of all components, the entire system configuration can be graphically displayed at a glance.

System Wide Automation

- Automation components have been developed to monitor the standardized messages and evaluate the context and state of specific components as well as the system in general. Upon the detection of specific conditions, associated predefined actions are automatically implemented.

Failover

- Component, computer, bus, and LAN failover can largely be handled by the middleware. Component-to-bus connections can be re-established transparently unburdening the components of unnecessary complexity (code development and testing).

Observed Benefits

- Significant reduction in integration time
- Components (new, upgraded, or redundant) and sub-nets can be added/removed with little to no impact for testing and parallel operations
- Small independent components (with no impact to other components) can be developed and added (with no impact to operations) for immediate benefit
- Ideal for multiple, small development teams to build and test applications rather than single large development teams

Exploded View of the Component-to-Message Bus Interface

