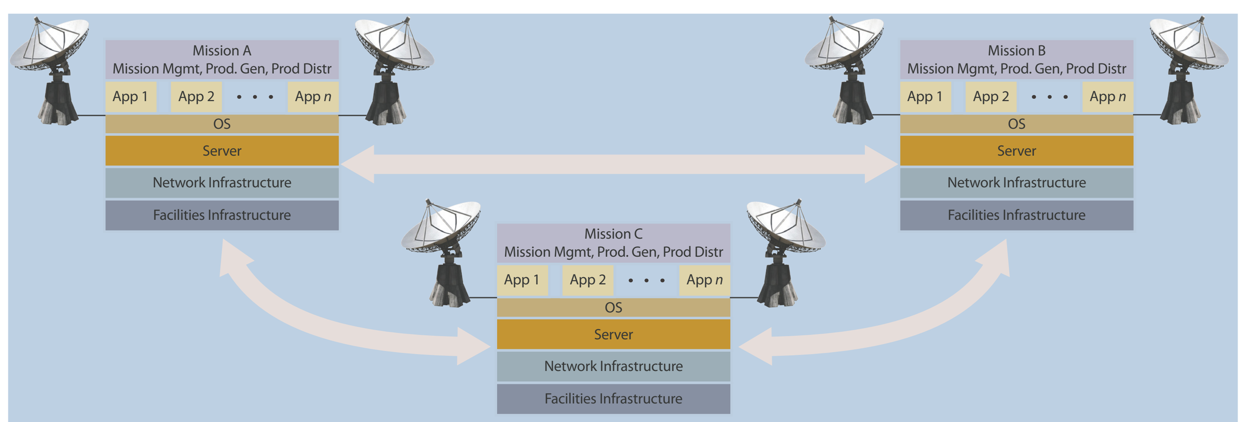
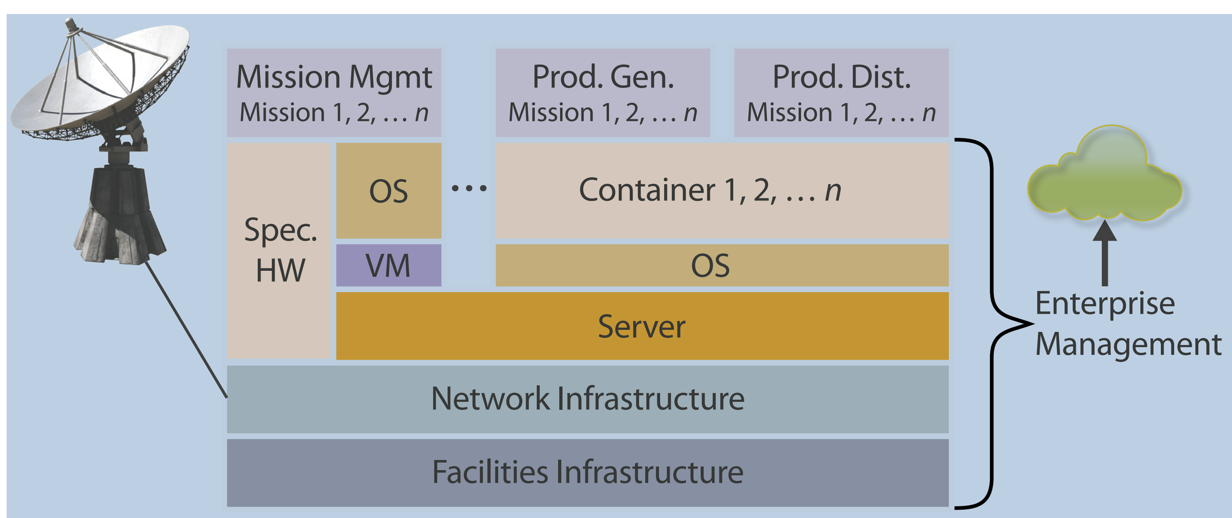
A New Approach to Sustain Legacy Satellite Ground Systems

Ye Men, Rachid Dahmani, Shahram Tehranian, Rob Haynes, Indy Saggu

Traditional Approach

- Traditional monolithic architected satellite ground systems operated with stove-piped model in a multi-missions environment
- Mission specific hardware dominated acquisition strategy during refresh and upgrade cycle
- Expensive procurement with long lead cycle and myriad of hardware platform, network, operating system, and applications to upgrade and update
- Difficult to maintain and operate over long lifecycle of missions as technology landscape evolve quickly
- Rigid hardware-based architecture style limiting interoperability among missions





New Approach to sustain legacy satellite ground systems

- Decomposition of hardware dominated ground systems (e,g, ingest, mission planning and mgmt, processing, product gen, and product distribution) into
  - Granular microservices oriented architecture
  - Microservices implemented with containers such as Dockers
  - Docker preserving binary interface and implementation of legacy mission applications with all its dependency rolled into a single cohesive unit of deployment
- Microservices managed by orchestration engines (Docker Swarm or Opensource Kubernetes) using rule-based capacity management in resource life-cycle management within Enterprise Management layer
- Separated control plane between mission management and production generation and distribution results in flexible system architecture, allowing
  - Mission management and product generation and distribution to each evolve on its timeline and needs
  - Incorporate technology appropriate of unique requirements of mission management vs product generation and distribution (Stringent Uptime vs Big Data)
- Containers are natural extension and first class citizen to Cloud to create hybrid infrastructure of on-premise and Cloud

Conceptual Implementation to a Small Scale Satellite Ground Segment (e.g, NJGS)

NJGS Overview

NOAA, in partnership with EUMESAT and NASA/JPL, has been operating the NOAA Jason Ground System (NJGS) to support the operation of Jason-2 and Jason-3 satellite missions. Jason-3 is the follow-on for Jason-2, with NOAA using a microwave radiometer, precision orbit determination components (e.g,, Global Positioning System (SPS), launch services, associated engineering services, the Jason-2 command and control, and data processing capabilities. EUMESAT provides spacecraft, altimeter instrumentation, precision orbit components, ground system and operations. NJGS has been operational and deployed at SOCC, WCDAS, FCDAS, and ESPC.

NJGS Functional Blocks Overview

Mission Management:

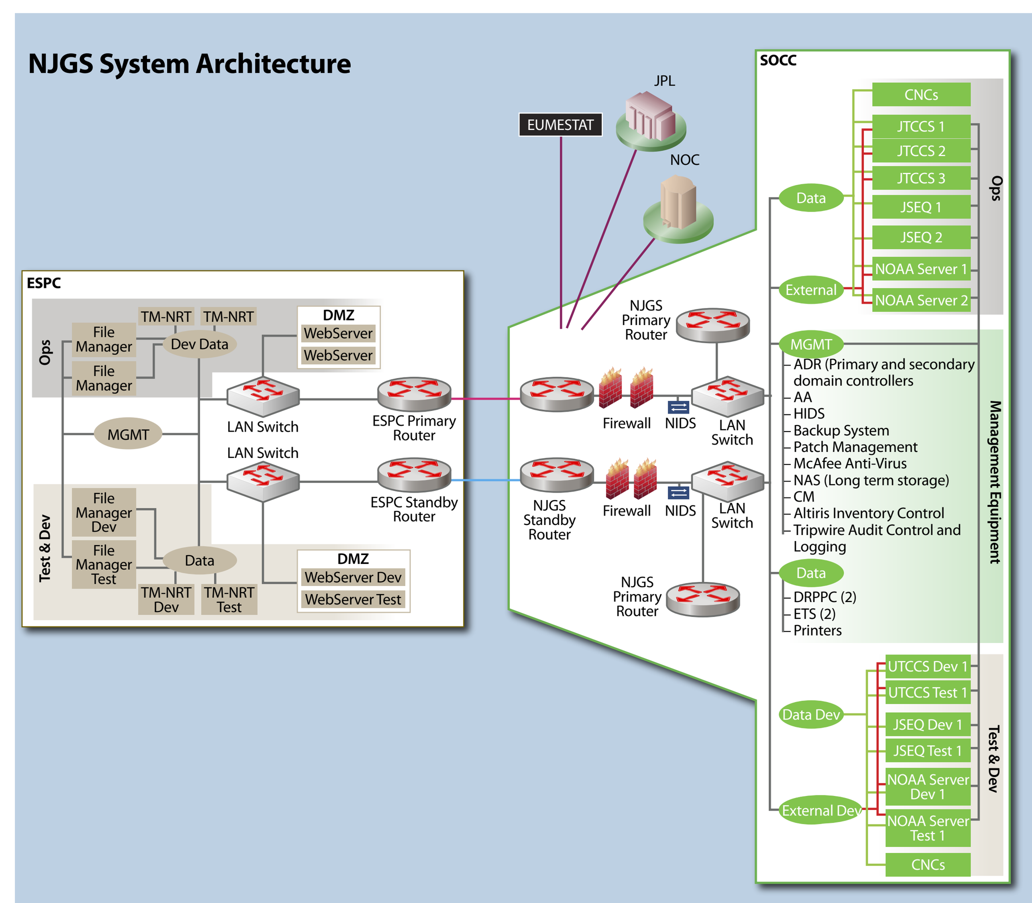
- JTCCS: Responsible for spacecraft and earth terminal command and telemetry processing
- NOAAServer: Responsible for file processing
- JSEQ: Responsible for schedule and command processing
- CIG: Responsible for telemetry ingest and processing and command transmission
- CNC: Provides operator access to and control of other subsystems
- DRPPC: Provides telemetry analysis and monitoring
- ETS: Provides office line analysis and engineering tools

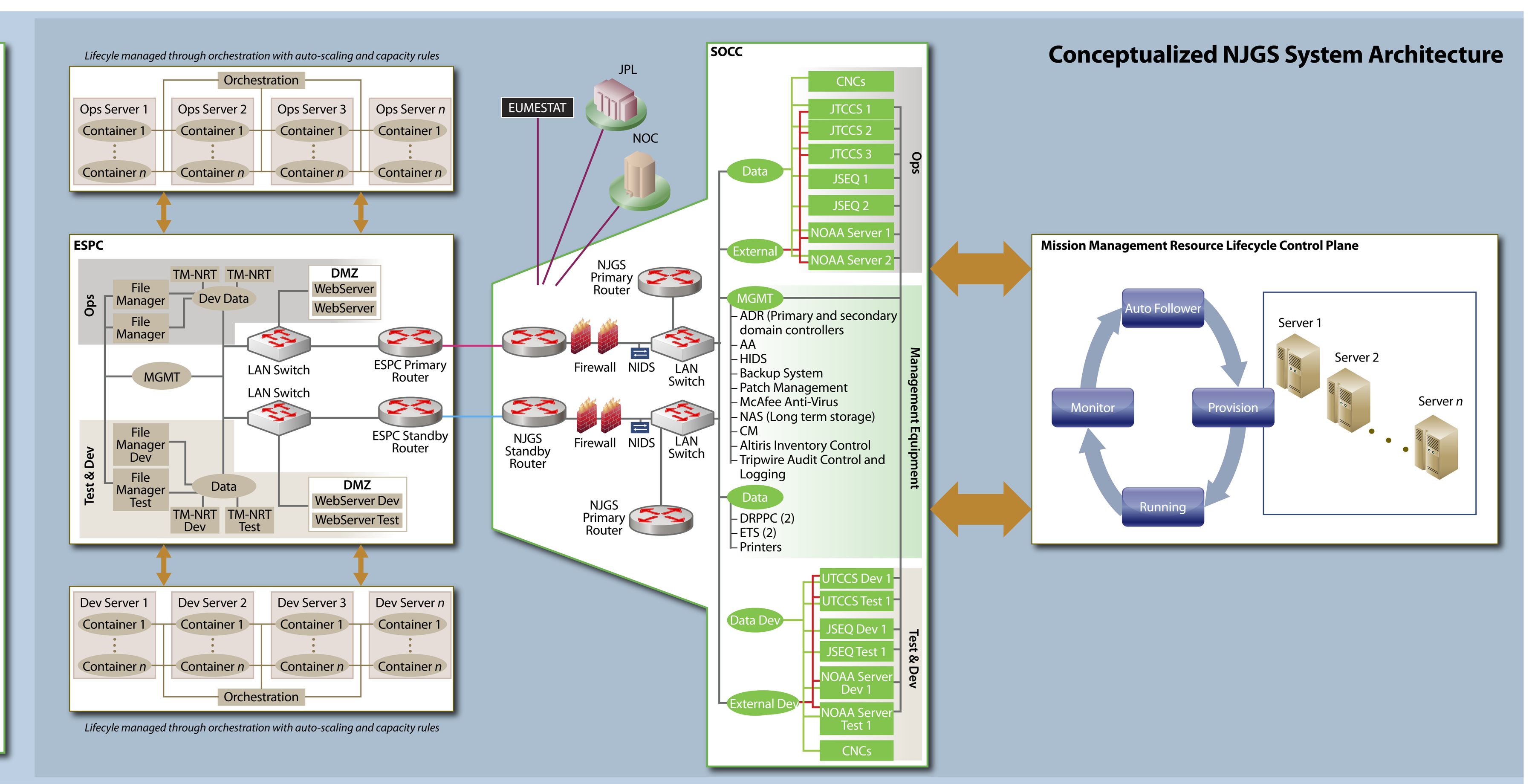
Product, Data Generation and Distribution:

- FileManager: Generates various user products, and provides file collection and distribution between internal and external ESPC systems
- WebServer: Host a web site containing composite logs generated by FileManager, hosted the NRTAVS website, displays the LogMonitor application for visualization of log errors
- TM-NRT: Generates high level products (OGDR) from raw data (LTM)
- ESPC workstations: Monitor and control the status of FileManager and WebServers

Idea Refresh Implementation:

- Separation of control plane between mission management and product data generation and distribution
- Mission management resource lifecycle via Virtual Machines using affinity and anti-affinity rules
- Product, Data resource lifecycle via Container orchestration using capacity rules and auto scaling
- Virtual Desktop Infrastructure to streamline and simplify O&M lifecycle of applications
- Potential reduction of 30+ servers to 10-





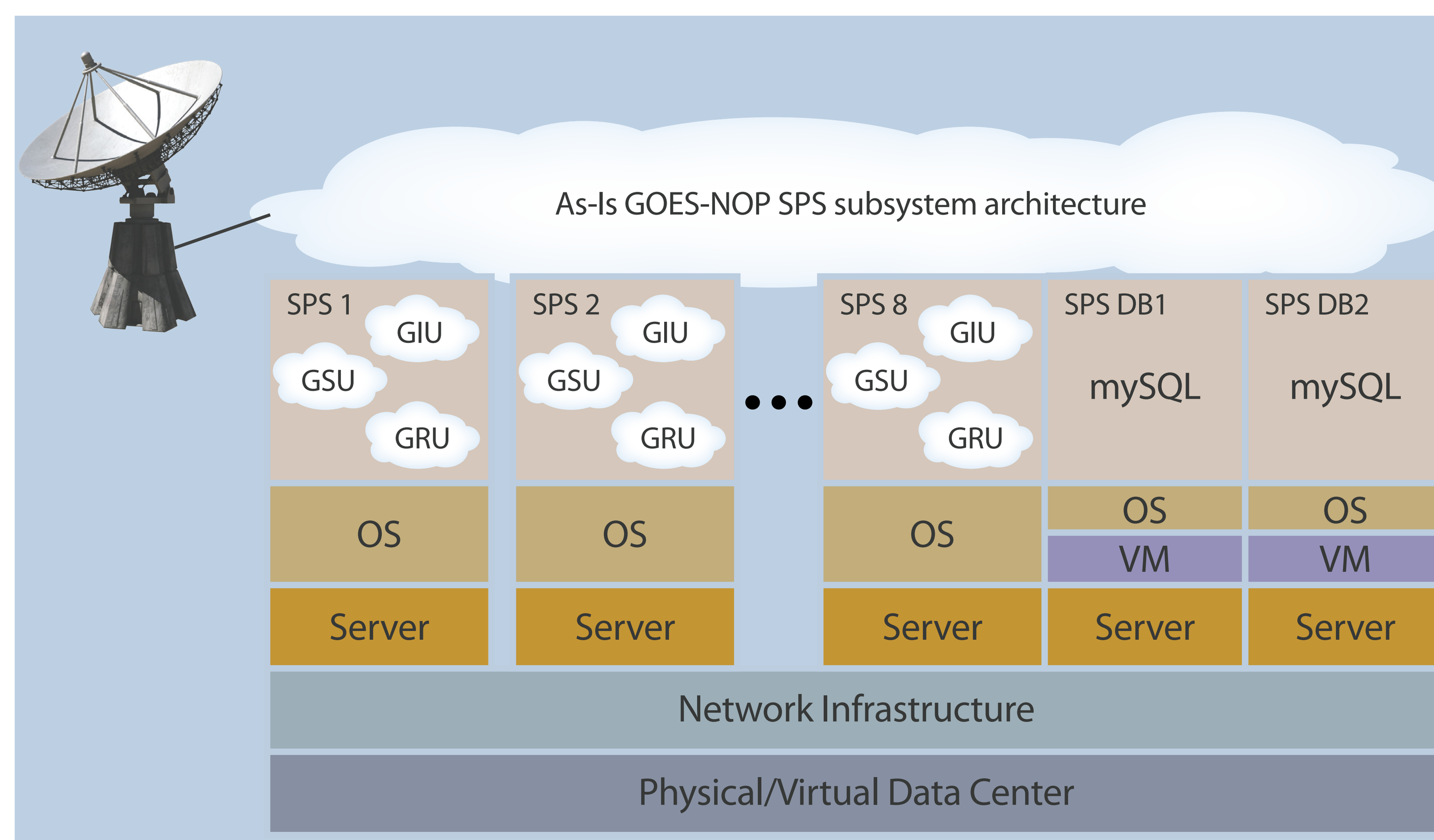
Conceptual Implementation to a Single Subsystem Within Legacy Ground Segment (e.g, GOES-NOP SPS subsystems)

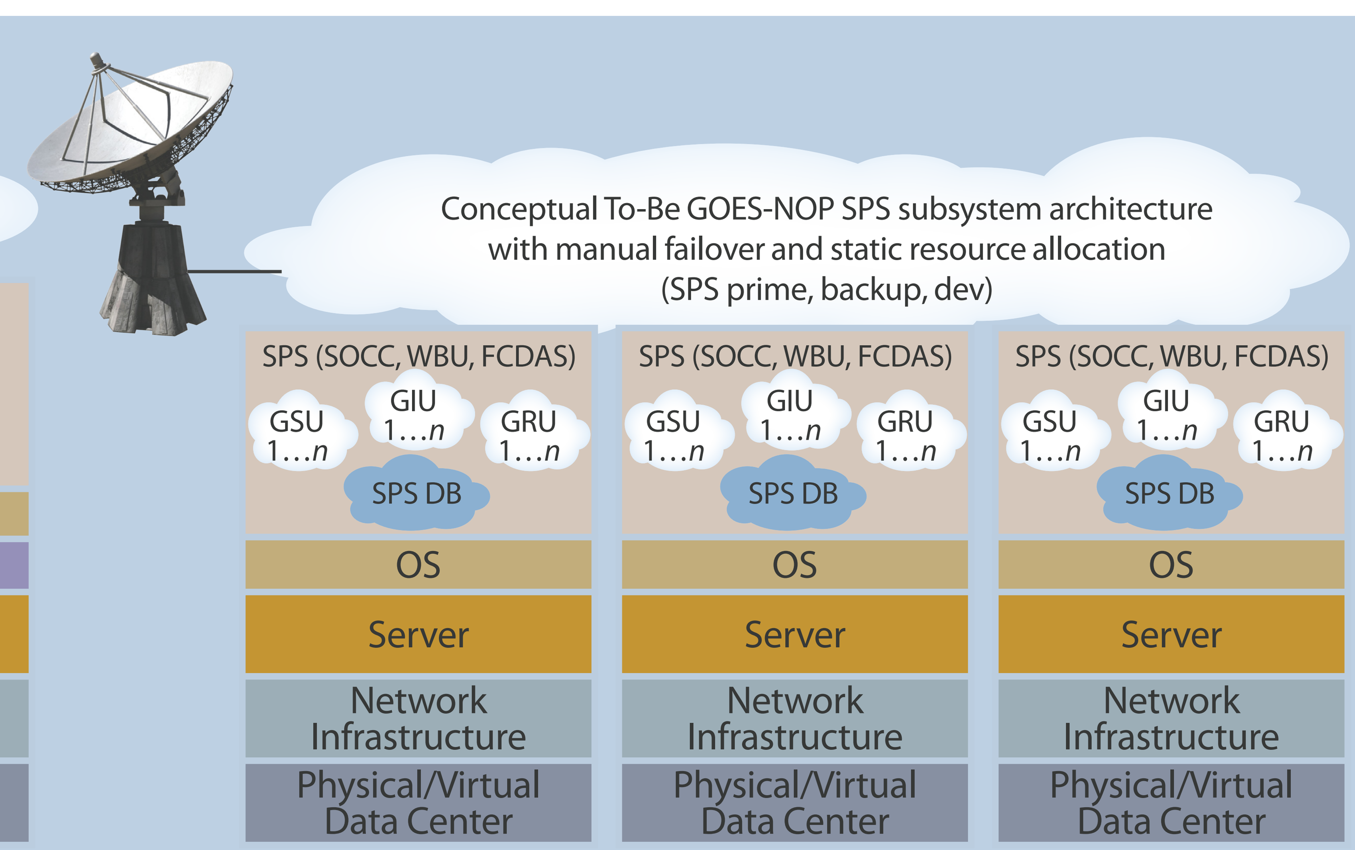
SPS Functional Overview

- A critical element of The GOES NOP Spacecraft Support Ground System (SSGS)
- SPS is responsible for the real time (rt) ingest and processing of the GOES sensor data to produce GVAR data stream. SPS also supports the Orbit and Analysis (O&A) determination function. The Sensor Processing System Database (SPSDB) maintains SPS initialization parameters on flat files and provides backend mySQL database storage/archival/query of satellite information such NLUT, CORT, and etc.
- Current operational deployment: 8 SPS and 3 SPSDB at WCDA, 1 SPS and SPSDB at FCDAS, 1 SPS and SPSDB at WBU.
- Current system architecture: Blade chassis with 2 blade servers and 2 expansion cards running Solaris 10 and zones providing all processing functions, GIU, GRU, GSU and SPSDB runs as separate VM co-located on other GOES-NOP servers

Ideal Re-hosted implementation

- Single physical server with RHEL 7.x hosting multiple SPS and SPS DB running Docker containers
- Up 6 EDT cards, pair of 3, co-located within one server to serve as conduit of GVAR data input and output
- GIU, GRU, GSU processing functions, implemented as Docker container, are stacked to form a single logical unit of SPS unit with SPSDB partitioned into standard Docker image of mySQL and custom legacy SPSDB Docker image
- Potential reduction of servers from 10 to 3.
- Legacy mission application interface and implementation persevered as it is with only change of deployment of unit.





ERT, Inc.
14401 Sweitzer Lane, 300
Laurel, MD 20707

www.ertcorp.com