Using Data Descriptions to Drive Enterprise Innovation

Donald Sather
Principal Engineer

March 2022

Approved for public release. OTR 202200022

Need for a New Approach

- Many organizations/agencies are evolving from traditional stovepipes toward a more enterprise paradigm to share data, provide enterprise situational awareness, and perhaps save costs
  - *Many of these involve "service" based architecture approaches*
- Difficult to develop truly "one size fits all" services due to unique mission requirements, satellites and concept of operations (CONOPS)
  - *CONOPS & Satellite often customized/optimized to meet mission requirements which often require unique ground capabilities*
- Expressing uniqueness in applications/services usually results in enterprise service "frameworks" that require extensive customization by the user
  - *Using a fixed set of enterprise service "frameworks" results in mission unique services which are as expensive to maintain as stovepiped services but have more uniform code quality*
- Bottom line: Uniqueness needs to be acknowledged and dealt with more efficiently by the enterprise and expressed somewhere other than the codebase

Can mission uniqueness be expressed somewhere other than in the application/service?

A Different Approach to Data

- Proposal
  - *Move mission unique aspects from the app/service to the data by using standard modelling language schemata centered around specific data types to describe mission uniqueness*
  - *Use data expressed in these standard descriptions to feed truly **generic** SATOPS functions (e.g. use of specific services to perform a function without the need for "modification" to meet unique program need)*
  - *Use mission planning and scheduling as a test vehicle as it is the area where mission uniqueness converges*
- Precedent
  - *Such standards already exist and are widely used for specific data types (XTCE (T&C - OMG), SSF (Contact Schedule Transfer - CCSDS), SMURF (Contact Schedule Request - CCSDS), etc.)*

A Different Approach to Data

Advantages & Disadvantages

- Advantages
  - *Provides unambiguous data standardization and understanding across the enterprise and mission areas*
  - *Provides a common method for mission partners or common ground resources to express use that applications/services can directly assimilate*
  - *Leverages model-based engineering practices many contractors already use for development*
  - *Reduces number of and simplifies common enterprise service software thereby reducing sustainment complexity and cost*
  - *Enables vendors/developers to focus on innovation of function vs. accommodating mission uniqueness*
  - *Can be turned over to a standards organization for sustainment*
- Disadvantages
  - *Schema take time (often years) to develop and mature*
  - *Existing programs or new methods of mission implementation may not be able to be mapped into the schema*
    - Schema may be able to evolve to accommodate “edge cases” – but it takes time
    - If they can’t be mapped then perhaps they truly need a unique service

Why Should a Data Centric Approach Succeed?

"Common Software/Services"

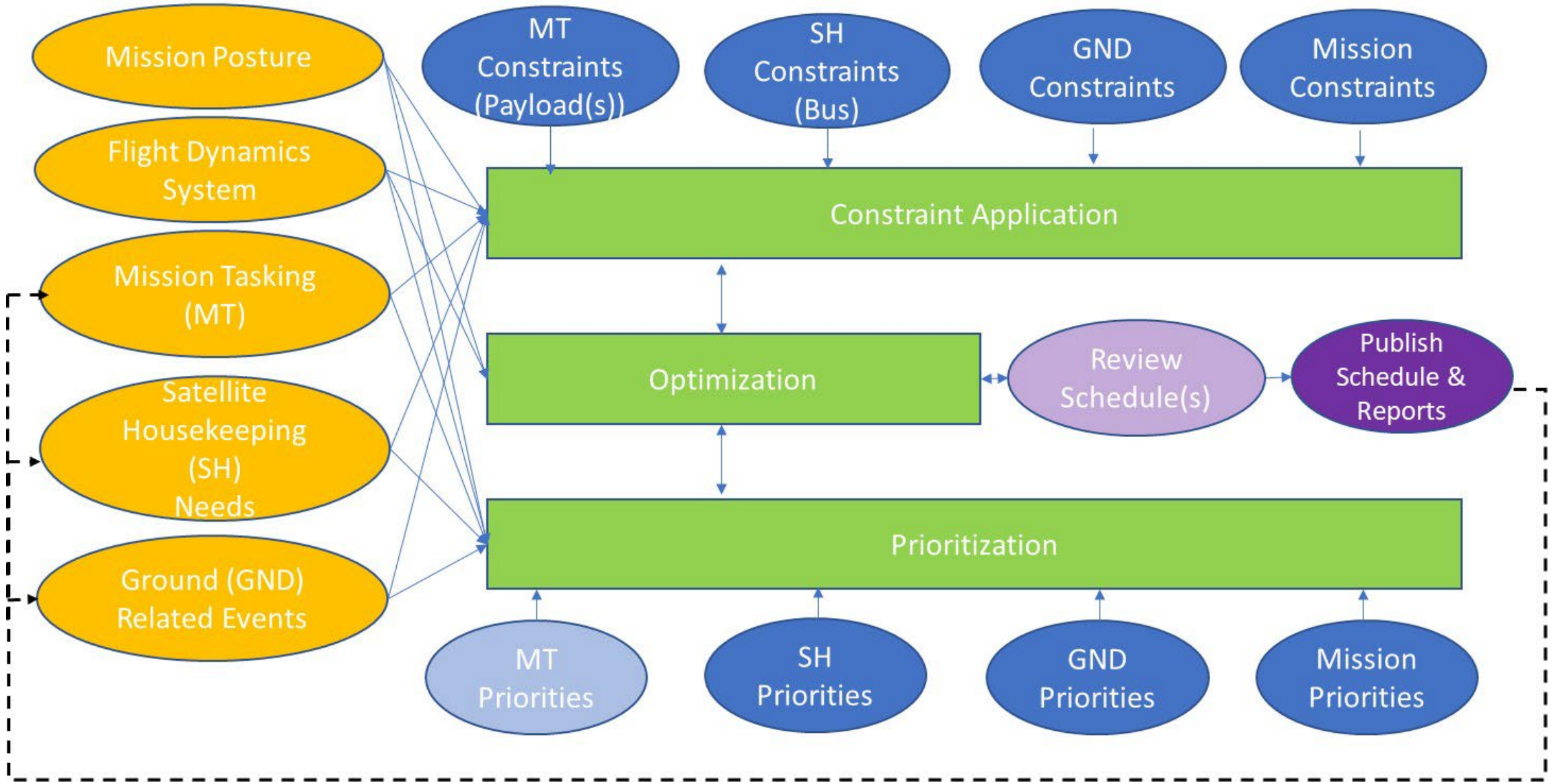
- "Common" software core "characterized" to meet mission unique requirements
- Tried in the past – failed
  - *MUS quickly outgrew common core*
  - *See Case Study in Backup*
- Changes in mission profile require changes to code and/or code associated databases
- MUS was the rule – one size didn't fit all
- Code changes (especially to common core) unresponsive to mission needs
- Enterprise sustainment and rollout coordination across programs difficult & expensive (see Case Study)

Data Centric

- Mission uniqueness expressed in the data as opposed to code
  - *Data is in a standardized schema of a high order modelling language*
- Changes in mission profile require changes to input data
  - *Mission controls/"sustains" the mission unique data*
- MUS is truly the exception
  - *Only if schema cannot accommodate*
    - Schema should evolve over time to catch the "corner cases" further lowering MUS
  - *Flying SPO responsible for maintaining their MUS (if any)*
- Enterprise sustainment more straight forward as it only covers truly common applications/services (no MUS)
- Sustainment of data standards can be turned over to standards organizations which the USG can have a voice in

Software is costly to develop, test, integrate and sustain – data is much less so

Notional Data Descriptions and Functions to Support MP&S



Summary

- Potentially high payoff leveraging existing design methods and products
  - *Use of standardized data description schemas based on a modelling language can allow mission uniqueness to be expressed in data as opposed to software*
    - Possible lifecycle cost savings by applying truly generic services to data descriptions
    - Data already must be maintained by a program
    - Modelling tools already embedded in most Contractors' developments – now just in a specified delivery format
  - *Descriptions are machine readable and unambiguous – less potential for interpretation error*
- MP&S is the single functional area where the most "mission uniqueness" is expressed – if it is successful here it probably can be more broadly applied

Way Ahead: Looking for funding to begin effort. Will report back next year

Backup

C^2 Evolution– A Case Study of Satellite Operations Center (SOC) at AFSPC/SMC

Common SOC Systems (1980-1996)

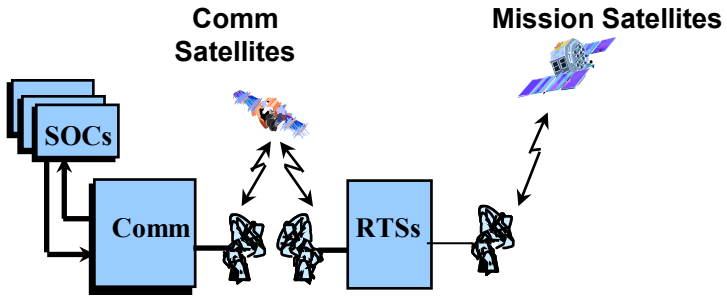
- Common systems (DSM, CCS) for TT&C
  - Evolved to multiple program-specific baselines
  - Mission unique functions difficult to maintain/upgrade and deploy across programs
- Dedicated ground systems used for payload mission data processing
- Replacement for DSM/CCS (SSCS) terminated
  - Not supported by business case
  - System too slow for mission timelines

"Stovepiped" SOC Systems (1997-Present)

- Each satellite program procures own TT&C system to replace CCS
  - Braxton based systems-GPS
  - ISI/Epoch based systems --MilSatComm
  - R&D missions on common systems ($2-5M/ sat for MUE/S)
  - R&D stand alone ($1.5-7M/ sat for MUE/S)
  - OS Comet based systems-OE, STPSat-1, STSS
  - Other systems are fully custom

Compatible SOC Systems (Future)

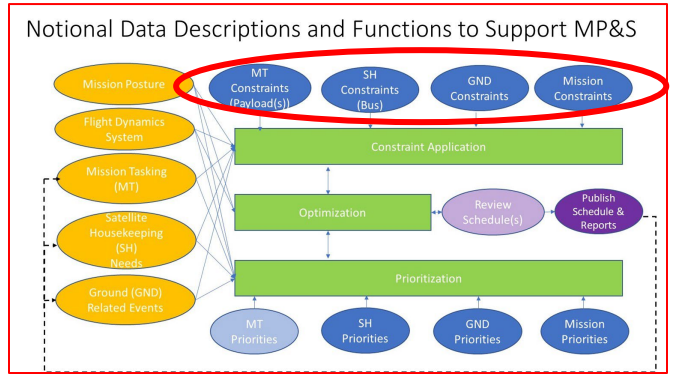
- Shared services & data
- Foundations being laid by EGS and SMC 2.0
  - Use modern IT and commercial practices - frameworks



Compatible SOC provides "middle ground" for reduced life-cycle cost and risk

Constraint Descriptions

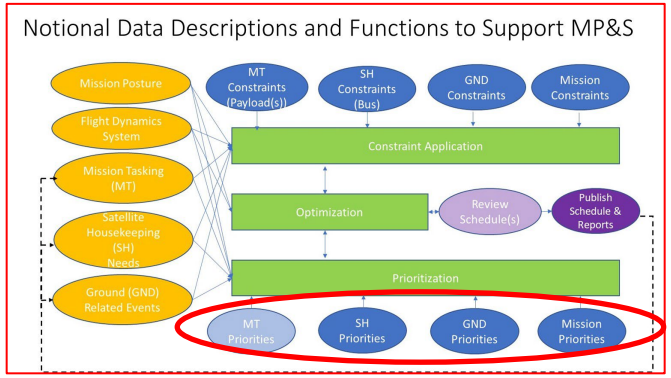
- Mission Tasking (Payload(s)): Physical constraints involving the payload(s)
  - *May involve interactions with the satellite bus*
  - *Includes payload resources & constraints including housekeeping (i.e. calibrations)*
- Spacecraft Housekeeping: Physical constraints involving the bus & tasks necessary to keep the spacecraft on-station & "healthy"
  - *May involve interactions with the payload(s)*
  - *Includes anomaly resolution constraints*
- Ground Constraints: All ground equipment/facility related physical constraints
  - *Includes contact schedule & related constraints*
- Mission Constraints: All mission related (typically CONOPS) constraints that affect how other constraint categories are applied or interact in certain mission conditions
  - *Some may be standing or may be for specific period or event depending on mission*
    - May require specify sub-category(ies) of constraints under Mission Tasking, Spacecraft Housekeeping & Ground to support different mission profiles
  - *Relationships with other missions*
  - *Mission permissions*
  - *May include personnel availability*
- Constraint attributes could include such things as:
  - *Constraint Malleability: Degree to which a constraint is permitted to be "bent"*
  - *Mission Posture: Mission posture to which the constraint is associated*



Perhaps these could be expressed in a single or individual schema(ta) – requires study

Priority Descriptions

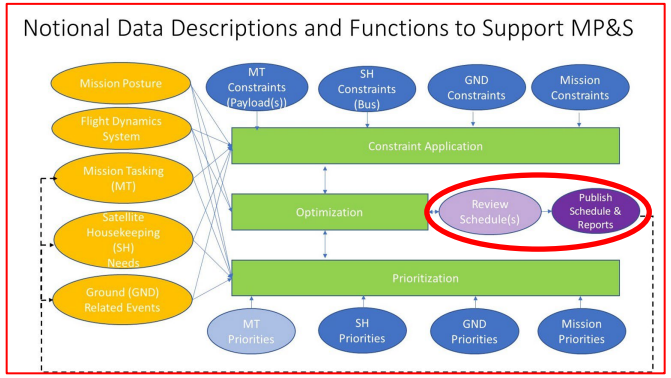
- Mission Tasking: Priority schema for mission tasks
  - *Shaded because this may be adjudicated by the tasking authority (i.e. the external tasking is received prioritized by another system/organization)*
- Spacecraft Housekeeping: Describes the relative priority of spacecraft Housekeeping (Health & Safety) tasks
- Ground: Describes the relative priority of ground maintenance and equipment availability
- Mission: How the other three areas interact with each other in the light of mission CONOPS in terms of priorities
  - *Tightly coupled with Mission Constraints*
    - Informs Mission Constraints of mission state



Could be viewed as another set of constraints or as part of optimization – study required

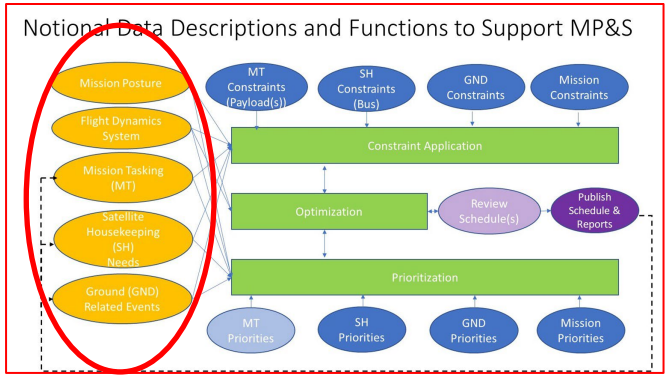
Schedule Descriptions

- Existing standard schema may exist for both Review Schedule and Publish Schedule
  - *CCSDS? – requires study*
- Review Schedule: Schema describes a schedule
  - *Allows a human review of a schedule or set of schedules*
  - *This is shaded due to CONOPS*
    - CONOPS may require a human or committee review schedule before publishing
    - Needed if mission planner alters optimization “weights” to produce different schedules to choose from
- Publish Schedule & Reports: Schema describes a schedule
  - *Made available for all authorized entities to review*
    - Maybe interim or final depending on CONOPS
    - if schedule is interim, tasking groups may re-adjust requests for the next schedule run
  - *Generates a standard set of reports (optional per CONOPS)*
  - *Publishes new contact requests if needed*
- Archival of interim products, schedules and reports can occur anywhere along the line



System Inputs

- Certain system inputs may require an “out of cycle” run of the MP&S cycle
  - *High priority tasking*
  - *Satellite contact schedule perturbations (missed contact)*
  - *Equipment failures/anomalies (space & ground)*
- Each group of system inputs should use a standardized schema
- Mission Tasking: Authorized user tasking
- Spacecraft Housekeeping Needs (Health & Safety): Spacecraft needs & events that effect P&S
  - *Includes spacecraft anomalies*
  - *“Spacecraft” includes payload(s) & bus*
- Ground Related Events: Ground events that affect P&S
  - *Can be facility or equipment related*
  - *Master Contact Schedule including bumps and outages*
  - *Missed or incomplete contacts*
- Flight Dynamics: Inputs from the mission’s flight dynamics system
- Mission Posture: Official mission posture



Data description standards need to be developed for system inputs also

Processes

Truly Generic

- Constraint Application: Applies constraints to system inputs in accordance with the individual sets as well as across sets in light of the Mission Constraints
  - *Produces many sets of constraint compliant "schedules" without regard to priority*
- Priority Prioritization: Applies priorities to system inputs in accordance with the individual sets as well as across sets in light of the Mission Constraints to feed into Optimization
- Optimization: Produces candidate schedule(s) based on constraint compliant "schedules" and relative priority
  - *Serves as HMI interface into the MP&S system*
  - *Allows mission planner to develop candidate schedules based on constraint "malleability" (e.g., how "hard" the constraint is)*
  - *Mission Planner may have the ability to "tweak" optimization based on criteria to choose "best" schedule Examples:*
    - Solely priority based (default)
    - Best fit for most requests fulfilled
    - Most efficient use of payload resources
  - *If mission unique optimization (i.e. something other than just strictly priority) would need to be specified in a standardized mission specific data description scheme for consumption by the Optimization service*
  - *Could be iteratively used to perform deconfliction via the use of standardized mission specific data descriptions*