Digital Twins Lifecycle Applications

Denver Smith, Digital Engineering Integration Office
Dr. Rob Stevens, Model Based Systems Engineering
Dexter Becklund, Vehicle Engineering Department

June 13th 2023

Agenda

- Digital Twin Lifecycle Architecture, Denver Smith
- Digital Twins for Spacecraft Concepts, Dr. Rob Stevens
- Aspen – Space Vehicle Digital Twins, Dexter Becklund

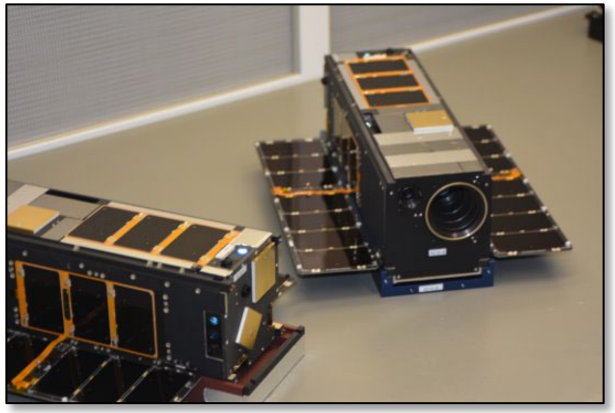
Digital Twin Lifecycle Architecture

Denver Smith,
Digital Engineering Integration Office

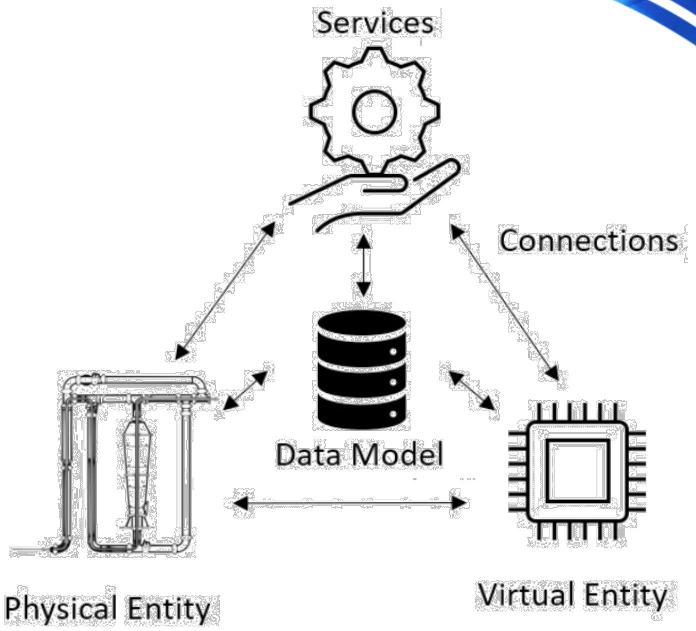
Digital Twin Ventures Project

Digital Twin Prototype and Specification

- Develop a 5D Digital Twin Specification
  - *Establish standard use cases across the lifecycle acquisition process*
  - *Develop architecture in support of digital twin*
  - *Aerospace Enterprise application for broad customer base*
- Prototype a Digital Twin
  - *Reverse engineer a digital twin for the Rogue CubeSat*
  - *Develop aspects of the digital twin at all phases within the lifecycle:*
    - Concept Dev, Detailed Design, Production & Testing, and Ops
- Partnered Enterprise Approach: iLab Sponsored, xLab and CSG partnered, broad ETG support and development



Establish a lifecycle perspective for a 5-dimensional Digital Twin



5-Dimensional Digital Twin Model

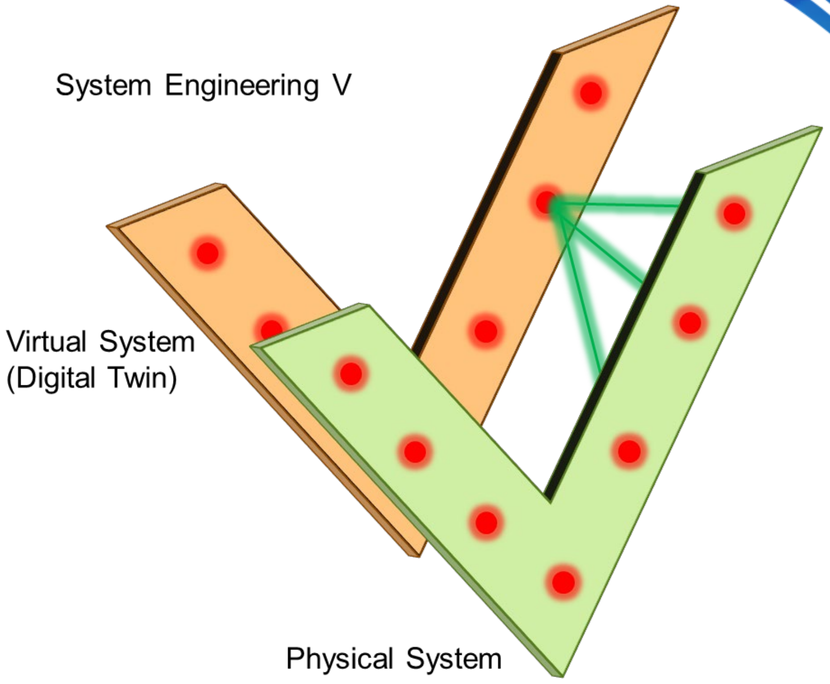
1. Physical Entity – the aspect of the system that exists in the physical world
2. Virtual Entity – sets of models that describe the physical entity
3. Digital Twin Data – the data associated with the system
4. Services – the functions associated with or governing of the digital twin
5. Connections – the bidirectional relationships between the other dimensions

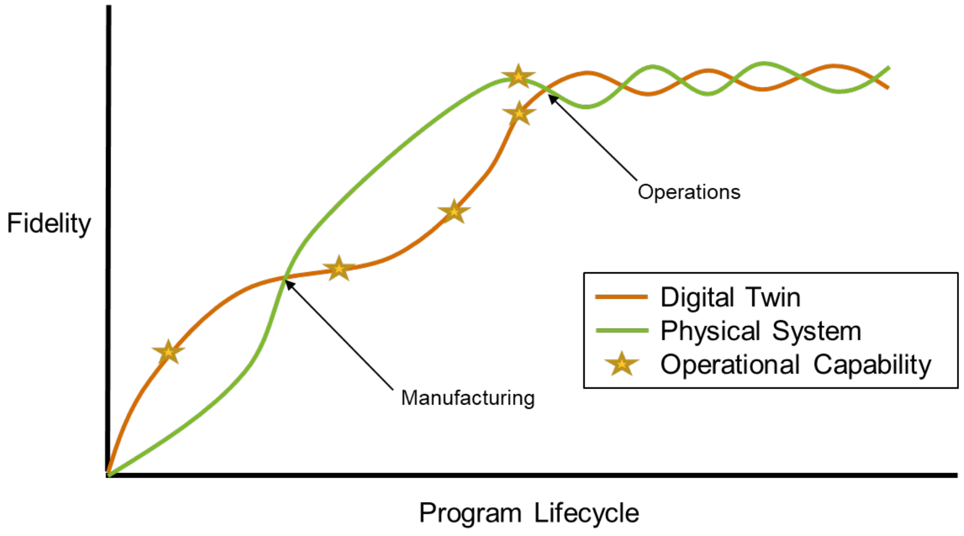
Digital Twin Driven Smart Manufacturing by Nee, Tao, & Zhang

Digital Twin Lifecycle

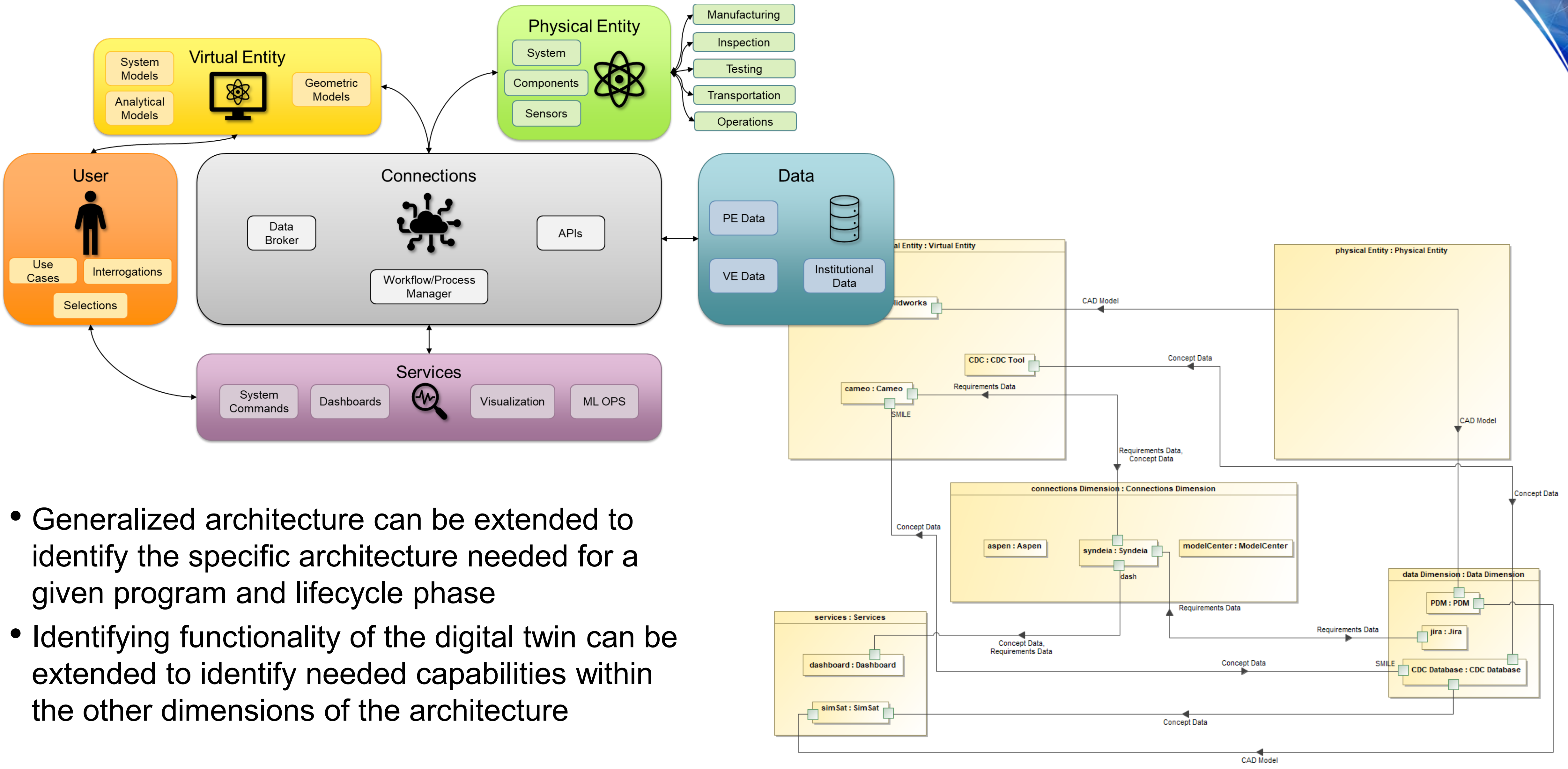
Digital Twin Prototype and Specification

- Move beyond the perspective of a digital twin as a singular model to a virtual system that matures in parallel with the physical system
- Establish standard development processes
  - *Use Case Development*
  - *Requirements*
  - *Design*
  - *V&V activities*
  - *Sustainment*
- Maturing through this process, the twin is both informed by and informs the development of the physical system across all stages of the acquisition lifecycle
- Several points along the lifecycle in which aspects of the twin must become operational





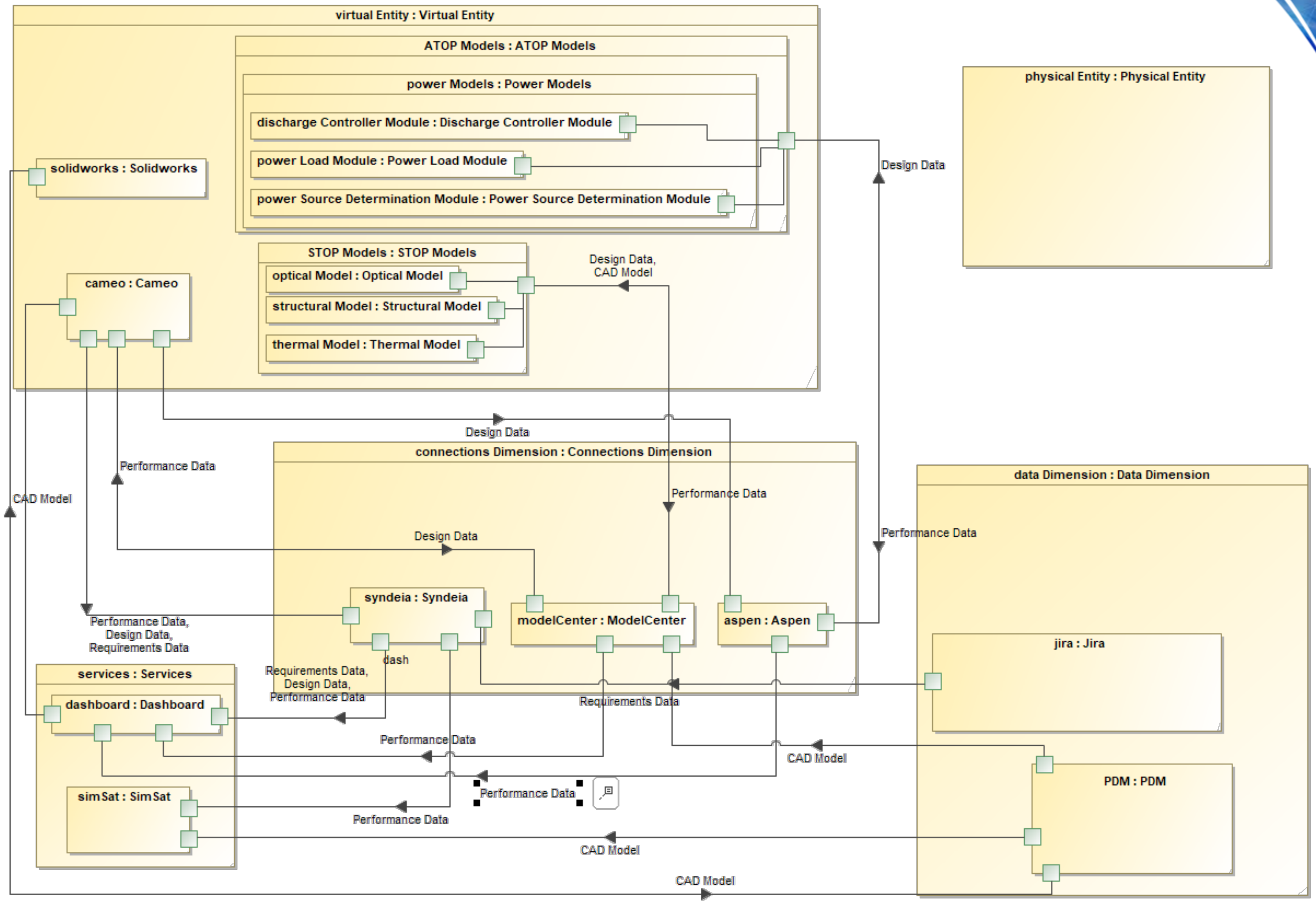
High Level to Specific Architecture – Concept Phase



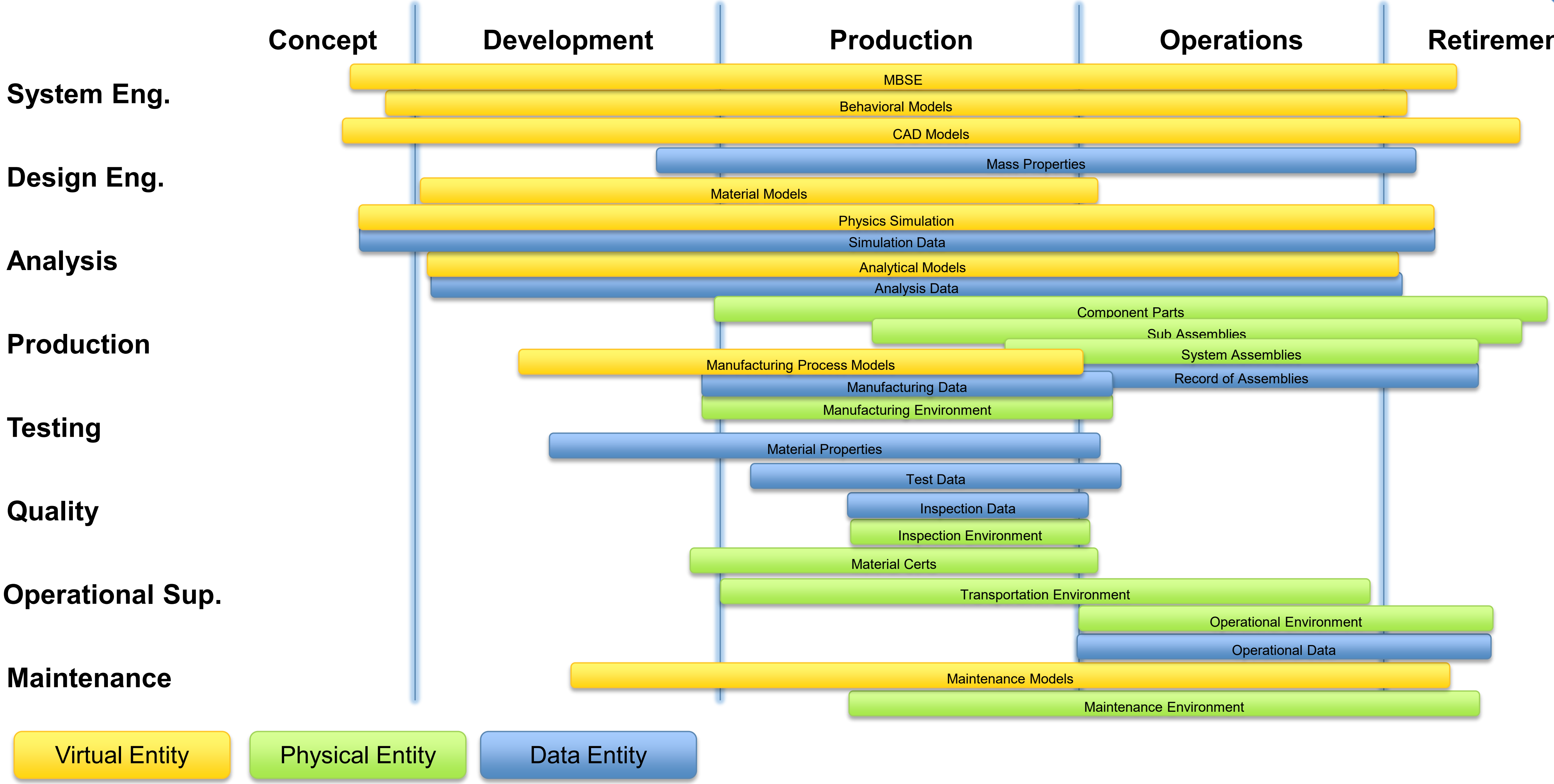
- Generalized architecture can be extended to identify the specific architecture needed for a given program and lifecycle phase
- Identifying functionality of the digital twin can be extended to identify needed capabilities within the other dimensions of the architecture

Rogue Digital Twin Architecture – Design Phase

- Data, Tools and processes are driven by the outputs of the digital twin
- Rapid expansion of tools and data sets as program matures
- Greater need to manage/automate workflow increases demand on workflow managers
- Increased complexity requires direct management of growth and activity.



Standard Lifecycle Processes



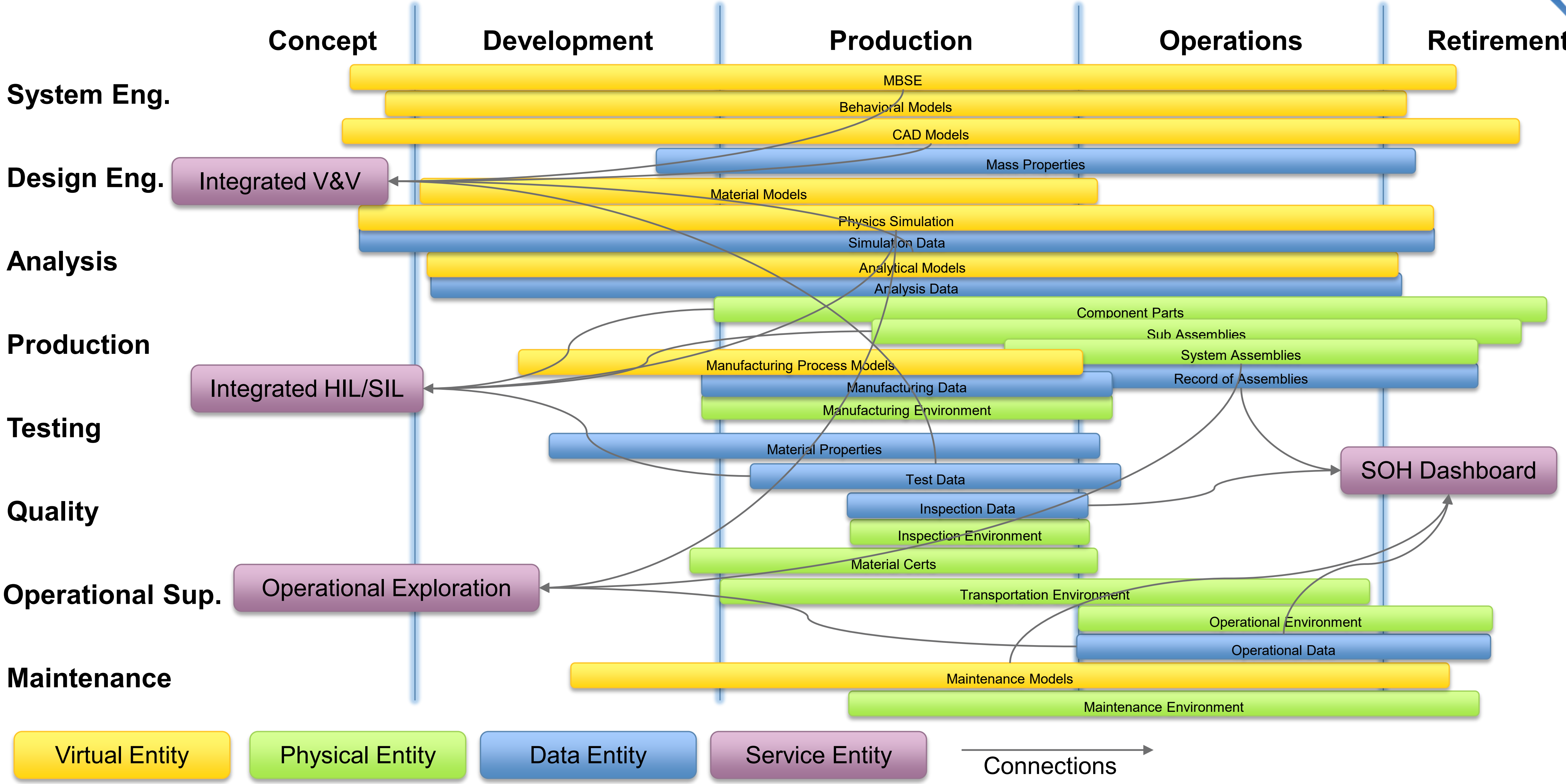
Concept
Development
Production
Operations
Retirement

System Eng.
Design Eng.
Analysis
Production
Testing
Quality
Operational Sup.
Maintenance

MBSE
Behavioral Models
CAD Models
Mass Properties
Material Models
Physics Simulation
Simulation Data
Analytical Models
Analysis Data
Component Parts
Sub Assemblies
System Assemblies
Manufacturing Process Models
Record of Assemblies
Manufacturing Data
Manufacturing Environment
Material Properties
Test Data
Inspection Data
Inspection Environment
Material Certs
Transportation Environment
Operational Environment
Operational Data
Maintenance Models
Maintenance Environment

Virtual Entity
Physical Entity
Data Entity

Digital Twin Services within the Lifecycle



Concept

Development

Production

Operations

Retirement

System Eng.

MBSE

Behavioral Models

CAD Models

Design Eng.

Integrated V&V

Mass Properties

Material Models

Physics Simulation

Simulation Data

Analysis

Analytical Models

Analysis Data

Component Parts

Sub Assemblies

Production

System Assemblies

Manufacturing Process Models

Integrated HIL/SIL

Manufacturing Data

Record of Assemblies

Manufacturing Environment

Testing

Material Properties

SOH Dashboard

Test Data

Inspection Data

Quality

Inspection Environment

Material Certs

Operational Sup.

Operational Exploration

Transportation Environment

Operational Environment

Operational Data

Maintenance

Maintenance Models

Maintenance Environment

Virtual Entity

Physical Entity

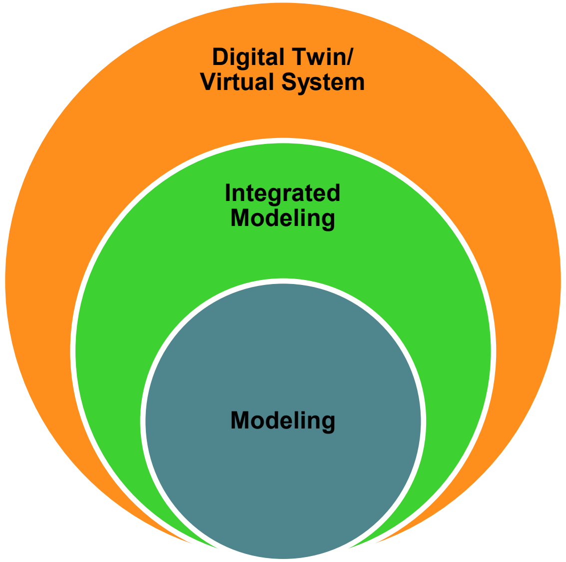
Data Entity

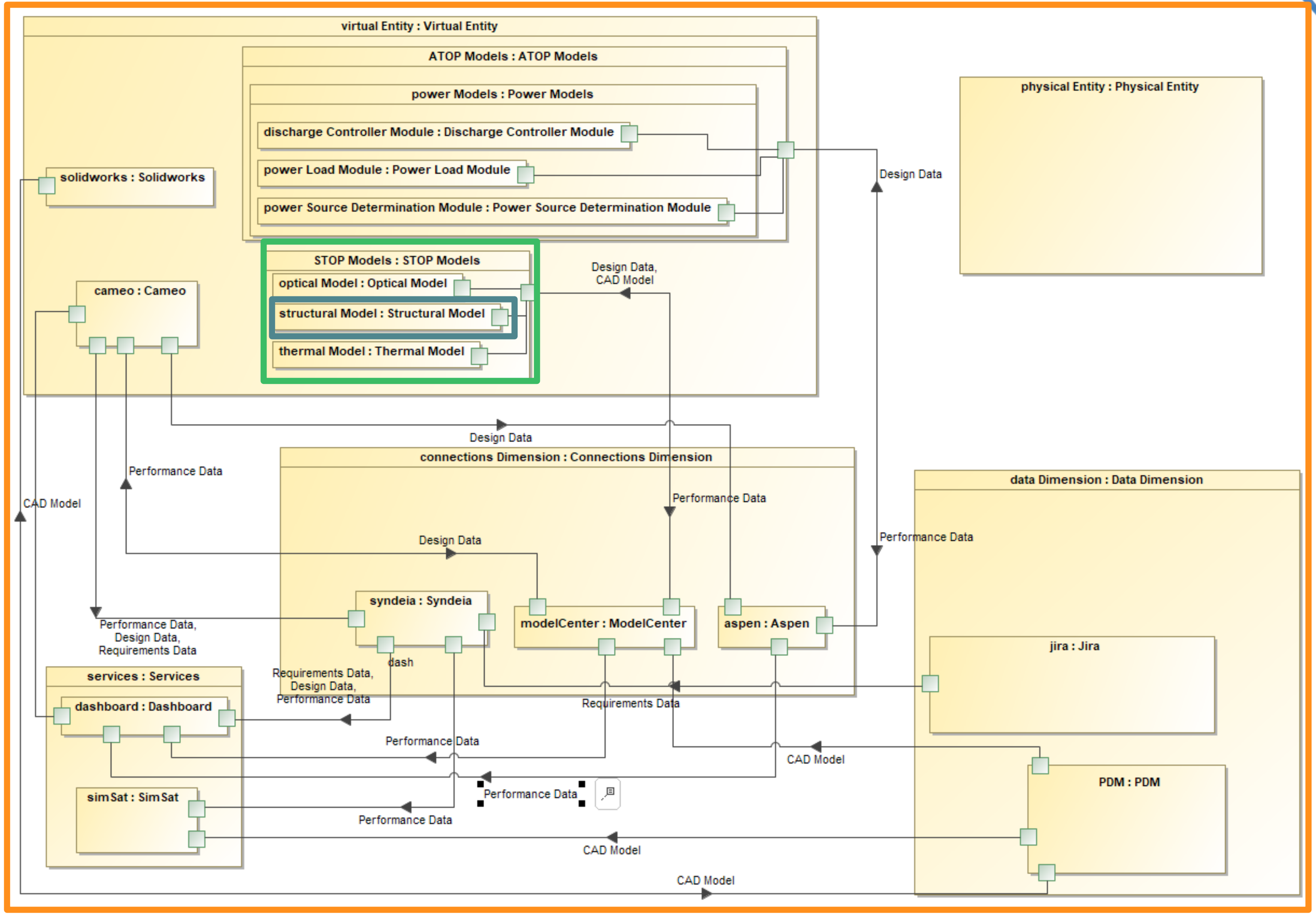
Service Entity

Connections

Digital Twin as a Virtual System

Early identification of use cases and establishment of requirements scopes the size and level of effort of your digital twin





Digital Twin for Spacecraft Concepts

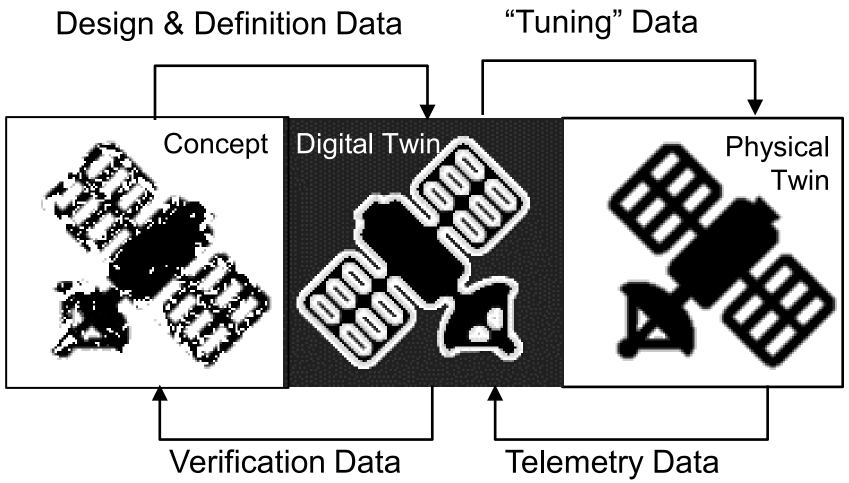
Dr. Rob Stevens,

Model Based Systems Engineering Office

What is a Digital Twin for Concepts?

- For operational spacecraft, data transfer occurs between the digital and physical twins
- During concept development, there is no physical system yet. Data is shared between the system concept design and the digital twin

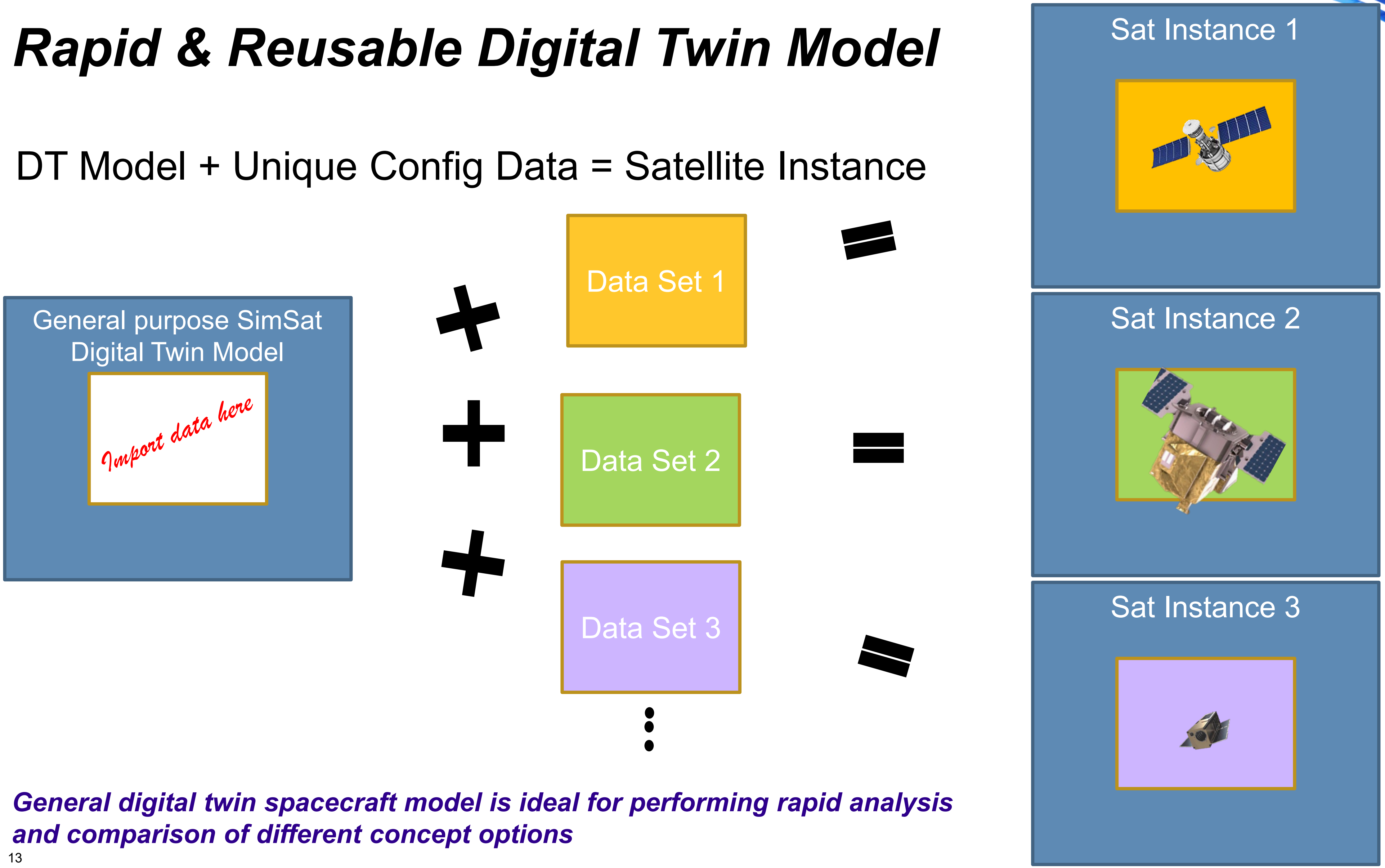
- The digital twin links verification data to system requirements



Early in the lifecycle, a digital twin is synchronized with system concepts. Later it synchronizes with its physical twin.

Rapid & Reusable Digital Twin Model

DT Model + Unique Config Data = Satellite Instance



General purpose SimSat Digital Twin Model

Import data here

\+ Data Set 1 = Sat Instance 1

\+ Data Set 2 = Sat Instance 2

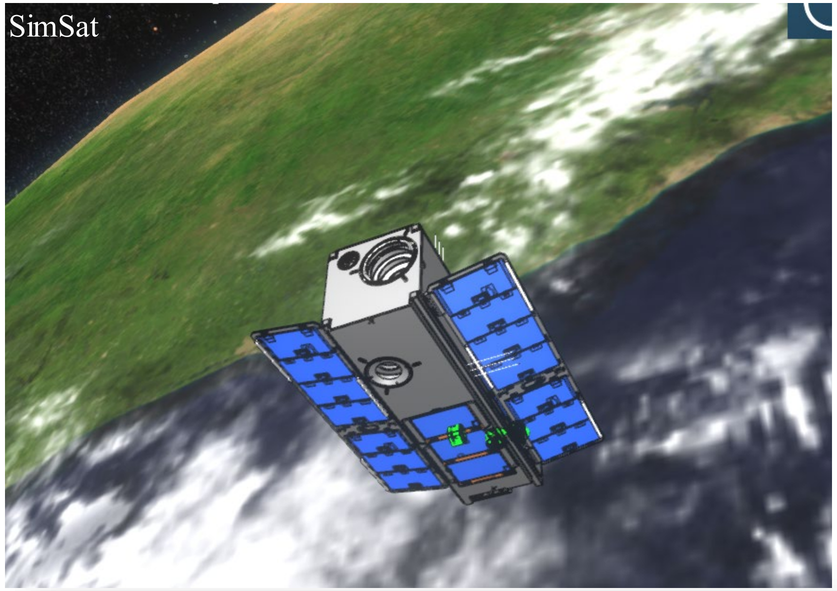
\+ Data Set 3 = Sat Instance 3

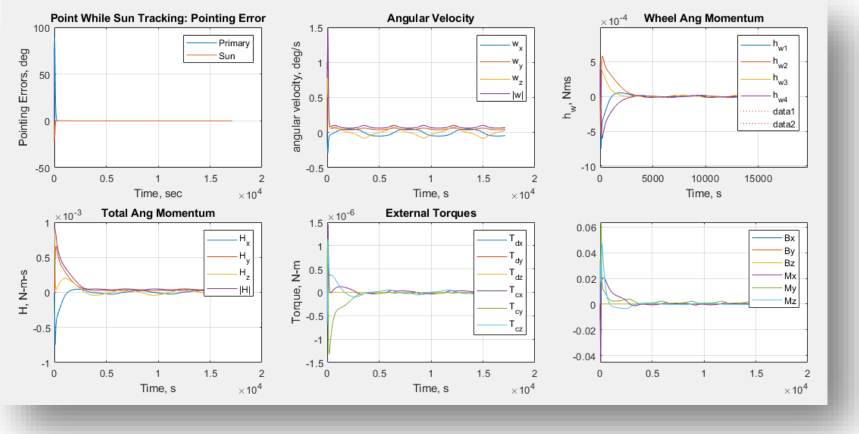
⋮

General digital twin spacecraft model is ideal for performing rapid analysis and comparison of different concept options

SimSat Characteristics

- Simulation and visualization environment for spacecraft systems
  - ***One-click access** from design data to sim environment. Allows rapid switching between vehicles for comparisons (trade analysis, eval architecture from different s/c perspectives)*
  - *High fidelity satellite behavior (RWs, mag torquers, PRY, power budget, state machines)*
- Uses *integrated* set of models vs. *federated* set of models
  - *Needed to integrate sim models suited to concept design, so created and connected tailor-made python codes*
- Focus is on **vehicle** performance
- Analysis data to support various activities throughout lifecycle
  - ***Concept Development**: Rapid trade space analysis, verify mission requirements*
  - ***Preliminary Design**: High fidelity simulation data to inform component selection. Verify system-level requirements.*
  - ***Detailed Design**: Verify component-level requirements and specs*
  - ***I&T**: Support test planning by informing test cases*
  - ***Ops**: Inform sat operators to improve performance or troubleshoot anomalies*
  - ***Disposal**: Generate ballistic coefficient data for deorbit/disposal estimation*

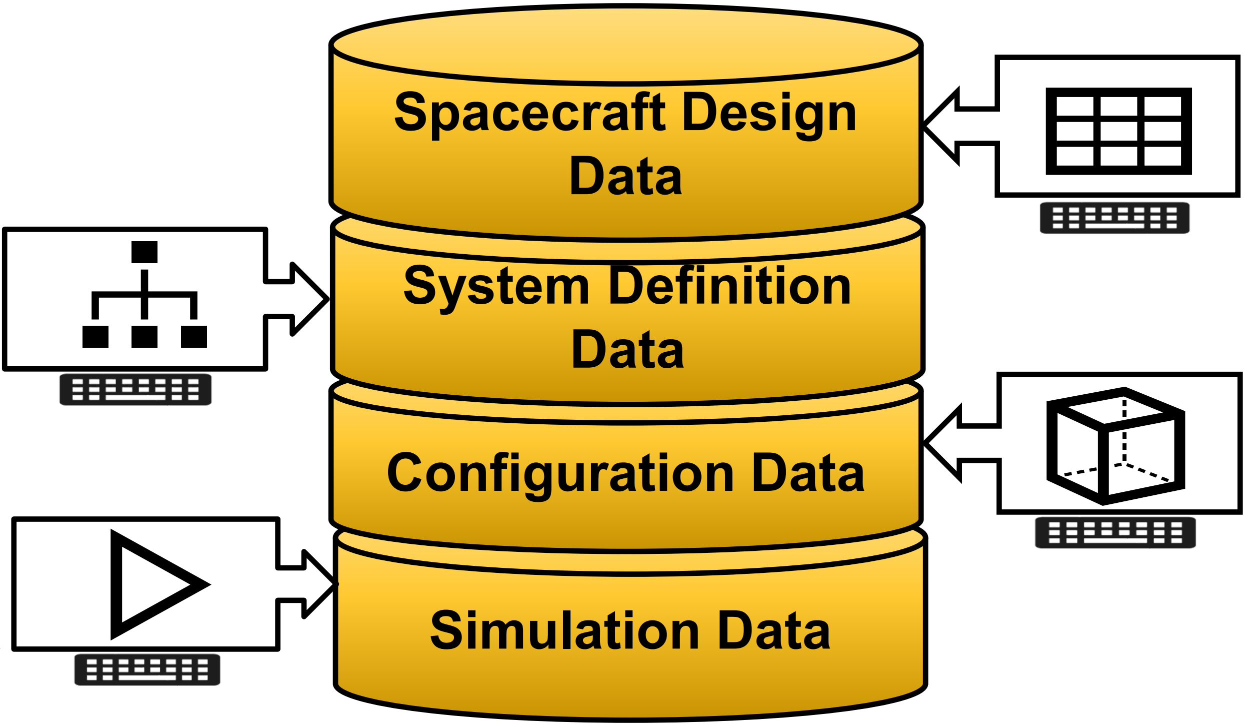




Attitude control performance

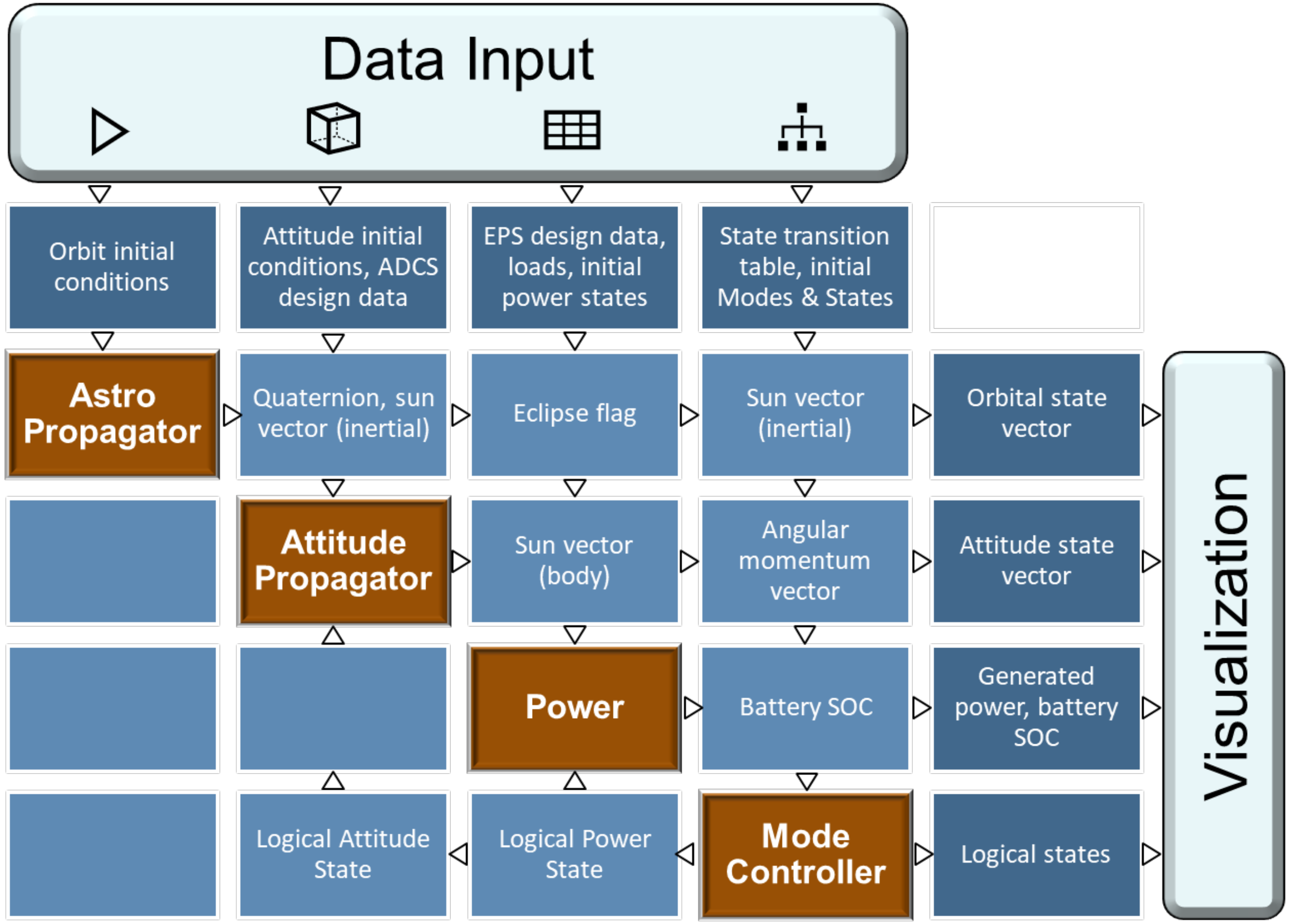
Data ASOTs and Pathways

- One key attribute of the digital twin flight simulator is that it is **connected to a data Authoritative Source of Truth (ASOT)**
  - *A general-purpose DT will ingest data to make an instance of a specific spacecraft*
- **4 types of data** for the DT simulator provided by 4 different tool sources
  - *Spacecraft Design*
  - *System Definition*
  - *Physical Configuration*
  - *Simulation Settings*
- Once data is captured in the ASOT database, external tools can access data to perform other analyses



Digital Twin Model and Data Framework

- Well-defined interfaces help achieve modularity and upgradeability
- N2 diagram shows givers and receivers of data
  - *Not intended to show actual data exchange between modules*
- Inputs come from the ASOT data base
  - *Users can adjust data as needed in the DT tool*
- Outputs provide visualization of behavior (animation, plots, etc.)



Orbit initial conditions	Attitude initial conditions, ADCS design data	EPS design data, loads, initial power states	State transition table, initial Modes & States	
Astro Propagator	Quaternion, sun vector (inertial)	Eclipse flag	Sun vector (inertial)	Orbital state vector
	Attitude Propagator	Sun vector (body)	Angular momentum vector	Attitude state vector
		Power	Battery SOC	Generated power, battery SOC
	Logical Attitude State	Logical Power State	**Mode Controller**	Logical states

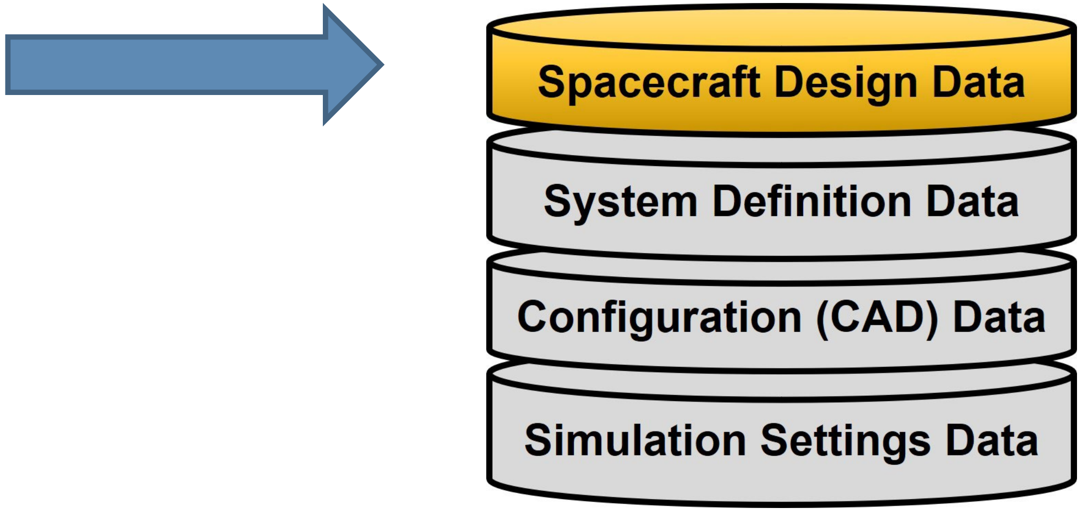
N^2 Diagram. *This diagram shows data exchange relationships between simulation modules, inputs, and output visualization. Different data types are represented by icons in the "Data Input".*

Spacecraft Design Data

Spacecraft Design Data includes properties such as

- Component size, mass, and power
- Total system mass properties
- Battery capacity
- Reaction wheel momentum capacity

Summary	Mass [kg]	Power [W]
Payload (with contingency)	**800.0**	**1000.0**
Uncontingencied Payload	300.0	400.0
Payload Contingency	30.0	40.0
Spacecraft Bus (with contingency)	**470.0**	**560.0**
ADCS	70.0	0.0
Power	50.0	25.0
Harness	40.0	-
Propulsion	20.0	8.0



Spacecraft Design Data

System Definition Data

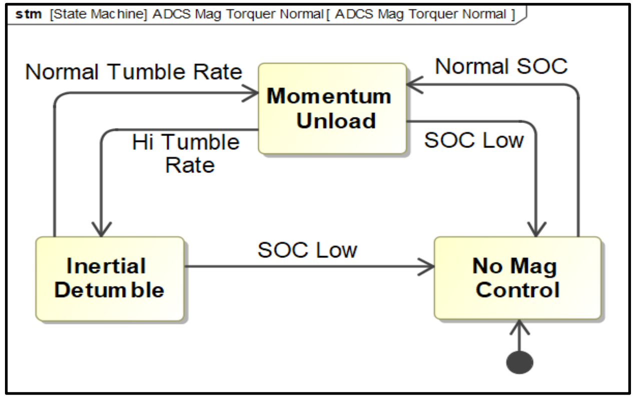
Configuration (CAD) Data

Simulation Settings Data

Concept design tools generate design data

System Definition Data

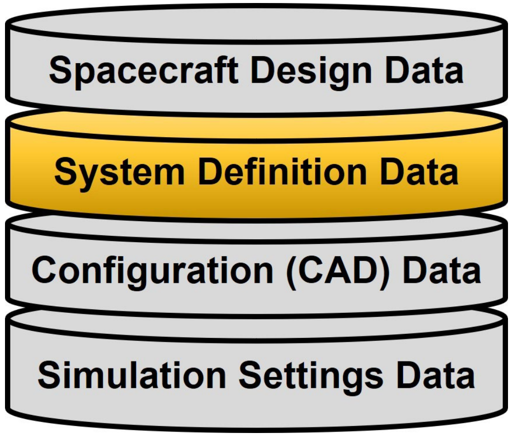
- System definition data may include performance requirements and desired control logic
- System logical behavior may be defined in a SysML state machine diagram
- Key attributes of the state machine are captured and structured in a the ASOT database
- Multiple state machines may be stored and simulated simultaneously in the virtual environment
- State machine data includes state names and transition conditions



State machine defined in SysML tool to capture logical behavior of a system

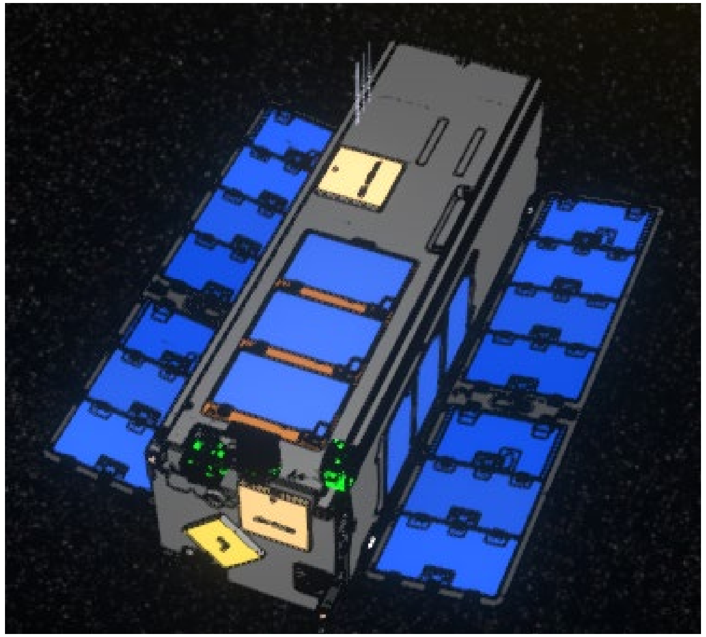
Current State		Next State		
		No Mag Control	Momentum Unload	Inertial Detumble
	No Mag Control	**No Mag Control**	Normal SOC	NA
	Momentum Unload	SOC Low	**Momentum Unload**	Hi Tumble Rate
	Inertial Detumble	SOC Low	Normal Tumble Rate	**Inertial Detumble**

State machines are broken down into their key attributes such as their states and transition conditions forming a state transition table

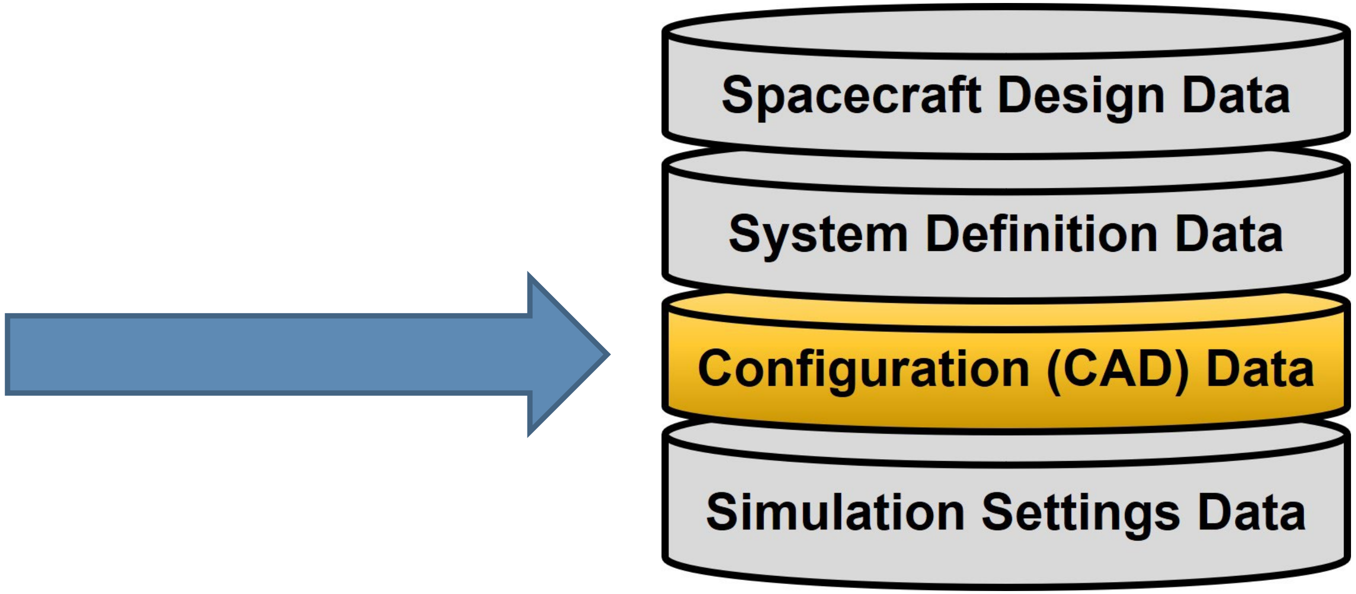


Physical Configuration Data

- Physical configuration in the form of CAD models
- Not stored in a database, but rather a library of CAD models



Physical configuration model created using CAD tool



Spacecraft Design Data

System Definition Data

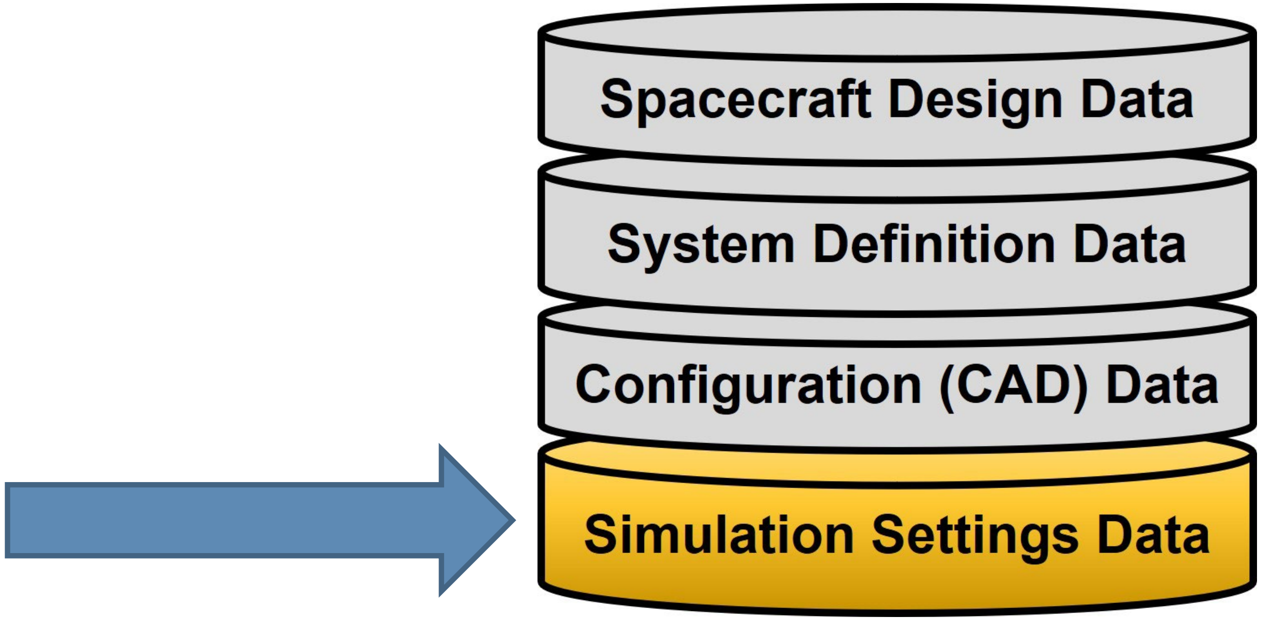
Configuration (CAD) Data

Simulation Settings Data

Simulation Settings Data

- The DT simulator itself will have settings that the user may want to store to repeat a simulation of a system of interest
- Simulation setting data may include initial orbit state, attitude and power control modes, control gains, and propagation parameters

Simulation settings set in simulation tool



Spacecraft Design Data

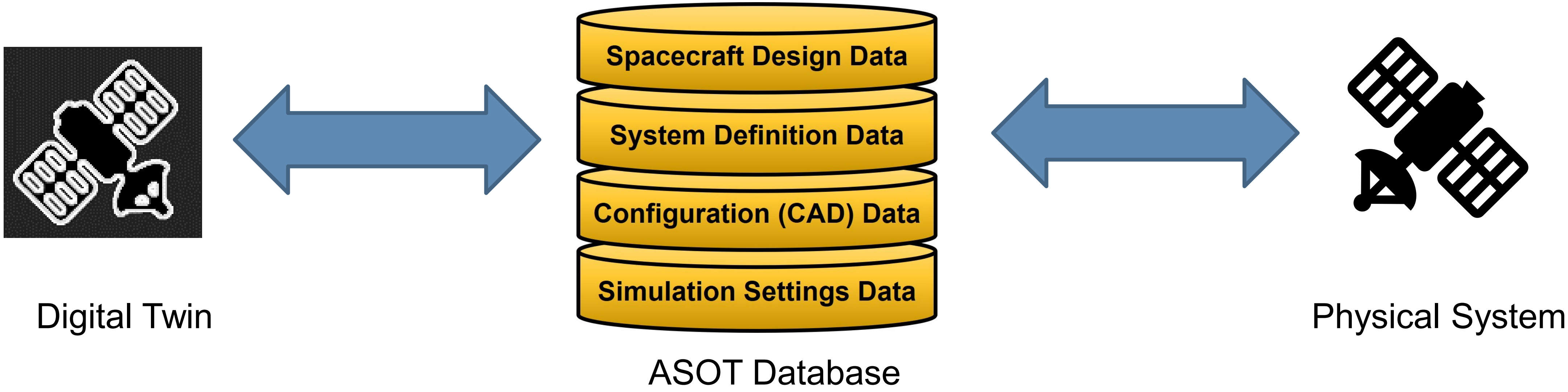
System Definition Data

Configuration (CAD) Data

Simulation Settings Data

Twinning with Physical Systems

- For concepts that will be built into operational physical systems, consider including “hooks” allowing model updates based on real telemetry data and TLEs
  - Ex. TLE sets provide real Ballistic Coefficient data that are difficult to estimate analytically
    - Useful for improving estimates for attitude disturbance torques for LEO satellites



Digital Twin

Spacecraft Design Data

System Definition Data

Configuration (CAD) Data

Simulation Settings Data

ASOT Database

Physical System

Summary

- A digital twin built to simulate spacecraft concepts in a virtual space environment will **verify design requirements** enabling **holistic analysis** of both logical control and physics-based behaviors
- The **data** necessary to run the simulation are generated by different tools and **reside in various ASOTs** for spacecraft design, system definition, simulation, and configuration.
- The **digital thread** weaving through the ASOTs and tools enable **data consistency** between the system’s requirements, design, configuration, and performance across its lifecycle.
- **Digital twin integration** starts in early **concept development** with established data pathways and well-defined schemas that enable engineers and stakeholders to interact with the synchronized data regardless of their tools.

Aspen – Space Vehicle Digital Twins

Dexter Becklund,
Vehicle Engineering Department

Aspen

Introduction

- Aspen is an integrated simulation approach to model a space vehicle in a flight-like environment – Digital Twins
  - *Goal is to virtually 'fly' a simulated satellite through various CONOPS and Design Reference Missions*
  - *Space vehicle components are simulated and/or emulated*
  - *Space vehicle environment and physics are simulated*
  - *Simulation component interfaces mimic flight-like interfaces*
    - Power system simulation interacts with other component simulations like hardware interacts with a power system
    - Simulation modules are broken into components to increase overall simulation modularity
  - *In the end, the simulation components look and feel like how a space vehicle is manufactured and integrated*

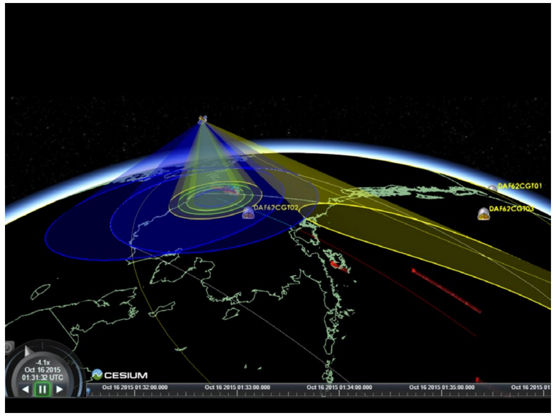
- Simulation modularity and well-defined interfaces between simulation components allows for increased simulation complexity down the line
  - *Simulation fidelity can be swapped in and out depending on the needs of the customer and conops*
  - *Swapping of simulated hardware with live hardware (for hardware-in-the-loop testing)*

- Simulation NRE will always exist when adding in new capability – the end goal is to have a library of simulation capabilities that can be easily swapped in and out
  - *Database of satellite components to rapidly execute studies of various satellite point designs*
  - *Software libraries to assist with other functionality such as visualization, MBSE interoperability, hardware connectivity*

Aspen provides a modular and extensible space vehicle simulation framework

Aspen Value

What does Aspen bring to the customer?



Example Cesium Globe View

- Aspen will ***output time-series data similar to an operational satellite would output telemetry***
  - *Simulated time-series state of health data, sensor readings, and other hardware telemetry from a bus or payload*
  - *“Truth” simulation data from physics or environmental simulation to compare against simulated hardware*
  - *Data can be post-processed to help V&V the mission against actual program requirements*
  - *Data will be visualized in industry standard tools such as Grafana*
  - *3D visualizations of the mission can be generated in common programs such as Cesium*
- Aspen ***gives program offices a way to virtually fly out their mission***
  - *Validation of concept designs through higher fidelity time-based analysis*
  - *Independent V&V of contractor modeling and simulation*
  - *Running mission scenarios for corner cases not included in original acquisition contract*
  - *Exploration and validation of contractor software and algorithms*
  - *V&V of requirements through analysis*
  - *Validation of command sequences to operational satellite*
  - *Assistance with anomaly resolution when the ‘physical twin’ is on-orbit*
- Aspen is being built for modularity and extensibility
  - *Initial Aspen federations will be for simple, single satellite missions*
  - *End goal is to have the capability to model multiple, cooperative missions*
    - Rendezvous and Proximity Operations
    - In-Space Assembly and Manufacturing
    - Edge processing, AI, Autonomy
    - Networking and Space Data Transport

Example Grafana Dashboard View

Aspen allows Aerospace to rapidly simulate missions and V&V against program requirements

Aspen Objectives

Overview

HIL/SIL/MIL Objective

- Aspen shall be capable of **running Hardware-in-the-Loop, Software-in-the-Loop, and Model-in-the-Loop testing** of a large range of systems that operate in and through the space domain.

Multi-Vehicle Mission Objective

- Aspen shall be **capable of testing missions involving multiple vehicles** to include the necessary functionality for multi-mission conops such as crosslinking, networking, RPO, etc.

Multi-Domain Objective

- Aspen shall be capable of **simulating, emulating, or interfacing with representations of air, sea, and land-based systems** that utilize and interact with vehicles in the space domain that are being tested in Aspen.

Autonomy V&V Objective

- Aspen shall be capable of **testing and performing V&V of autonomous mission concepts** to include, at a minimum, onboard autonomy (mission planning, C2, etc.), autonomous fault detection, onboard data processing, edge processing, distributed processing, etc.

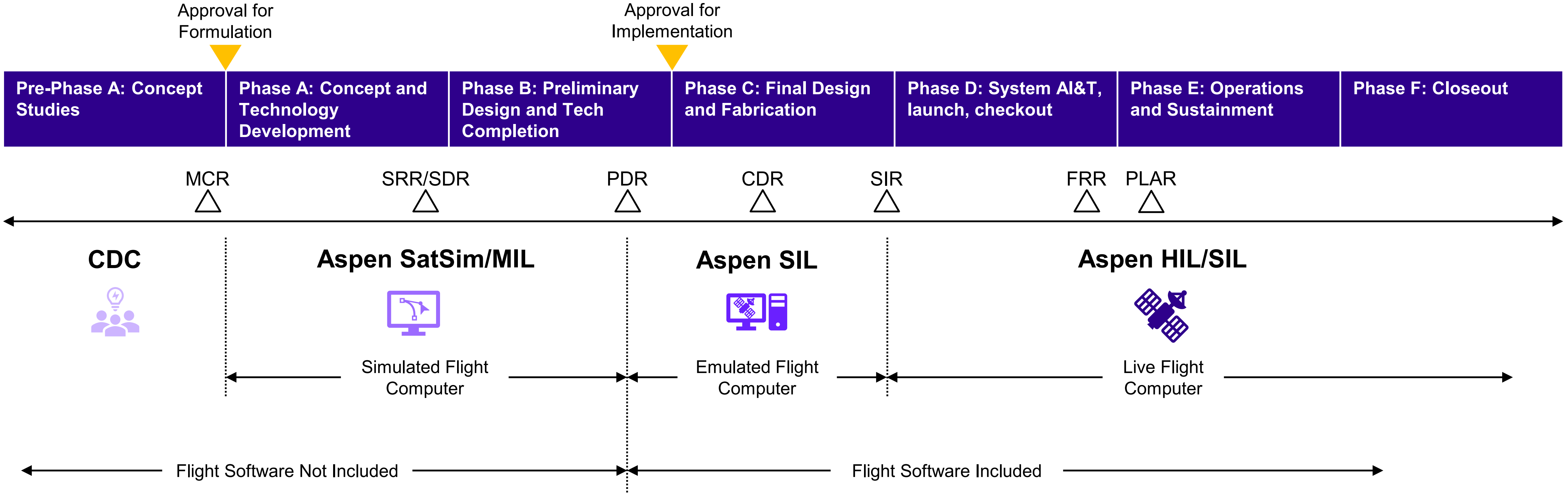
Digital Twins Objective

- Aspen shall be designed such that it can be **leveraged throughout the project lifecycle** from concept design to operations and disposal, able to support varying levels of model fidelity as well as flight-like software and hardware integration as needed throughout a project’s lifecycle.

Objectives satisfy higher level stakeholder goals of spacecraft hardware and software testing

Aspen Digital Twin

Usage across the acquisition lifecycle



Approval for Formulation

Approval for Implementation

Pre-Phase A: Concept Studies

Phase A: Concept and Technology Development

Phase B: Preliminary Design and Tech Completion

Phase C: Final Design and Fabrication

Phase D: System AI&T, launch, checkout

Phase E: Operations and Sustainment

Phase F: Closeout

MCR △ SRR/SDR △ PDR △ CDR △ SIR △ FRR △ PLAR △

CDC

Aspen SatSim/MIL

Simulated Flight Computer

Aspen SIL

Emulated Flight Computer

Aspen HIL/SIL

Live Flight Computer

Flight Software Not Included

Flight Software Included

Vehicle concept development, static designs, basic vehicle budget allocation (pointing, data, power, etc.)

CONOPS and Design Reference Mission (DRM) validation, dynamic mission analysis, higher fidelity budgeting, onboard spacecraft algorithm development

Software development and implementation, and higher fidelity mission modeling. Stress testing of algorithms and software to ensure functionality through edge cases.

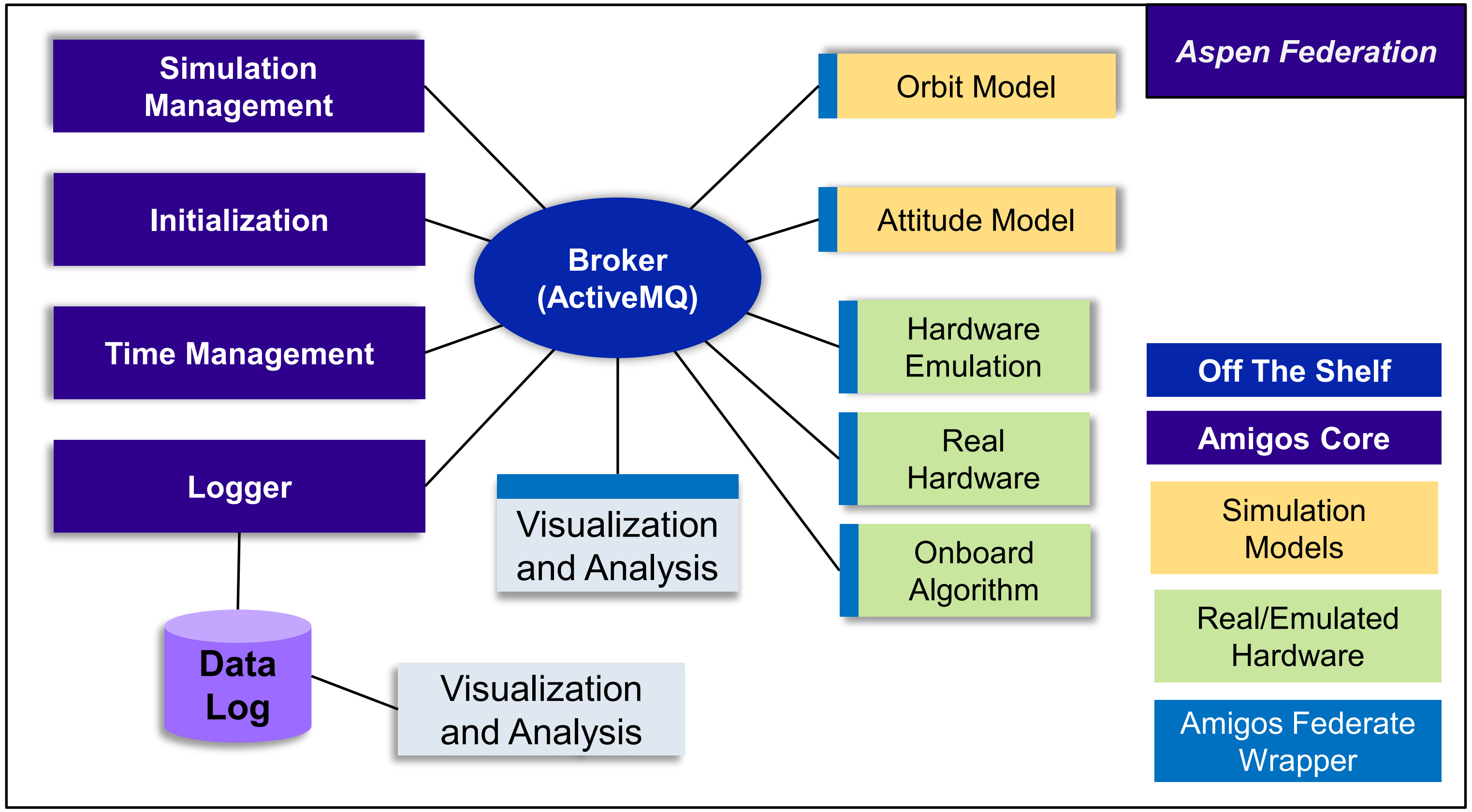
Final design implementation, final software development and integration with hardware, embedded system and FPGA implementation, out to flight

Increasing Model Fidelity and Complexity

MCR	Mission Concept Review
SRR	System Requirements Review
SDR	System Design Review
PDR	Preliminary Design Review
CDR	Critical Design Review
SIR	System Integration Review
FRR	Flight Readiness Review
PLAR	Post-Launch Assessment Review

Aspen Architecture

Using the Amigos federated mod/sim framework for space vehicle Digital Twins



Aspen Federation

Simulation Management

Initialization

Time Management

Logger

Broker (ActiveMQ)

Visualization and Analysis

Data Log

Visualization and Analysis

Orbit Model

Attitude Model

Hardware Emulation

Real Hardware

Onboard Algorithm

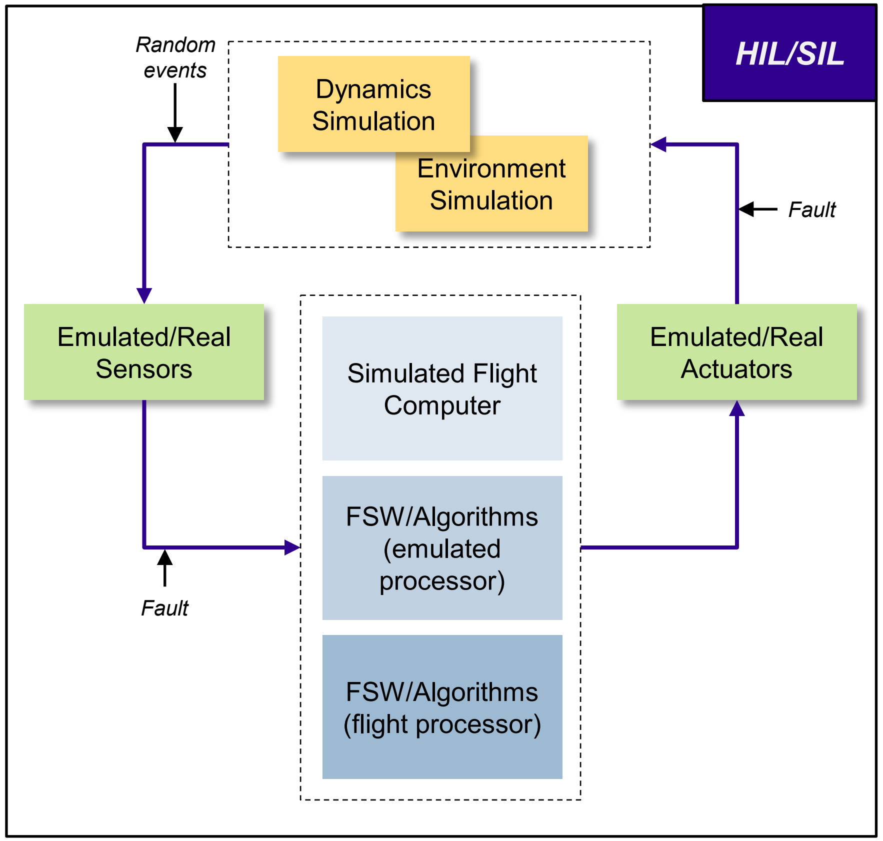
Off The Shelf

Amigos Core

Simulation Models

Real/Emulated Hardware

Amigos Federate Wrapper



HIL/SIL

Random events

Dynamics Simulation

Environment Simulation

Fault

Emulated/Real Sensors

Simulated Flight Computer

FSW/Algorithms (emulated processor)

FSW/Algorithms (flight processor)

Emulated/Real Actuators

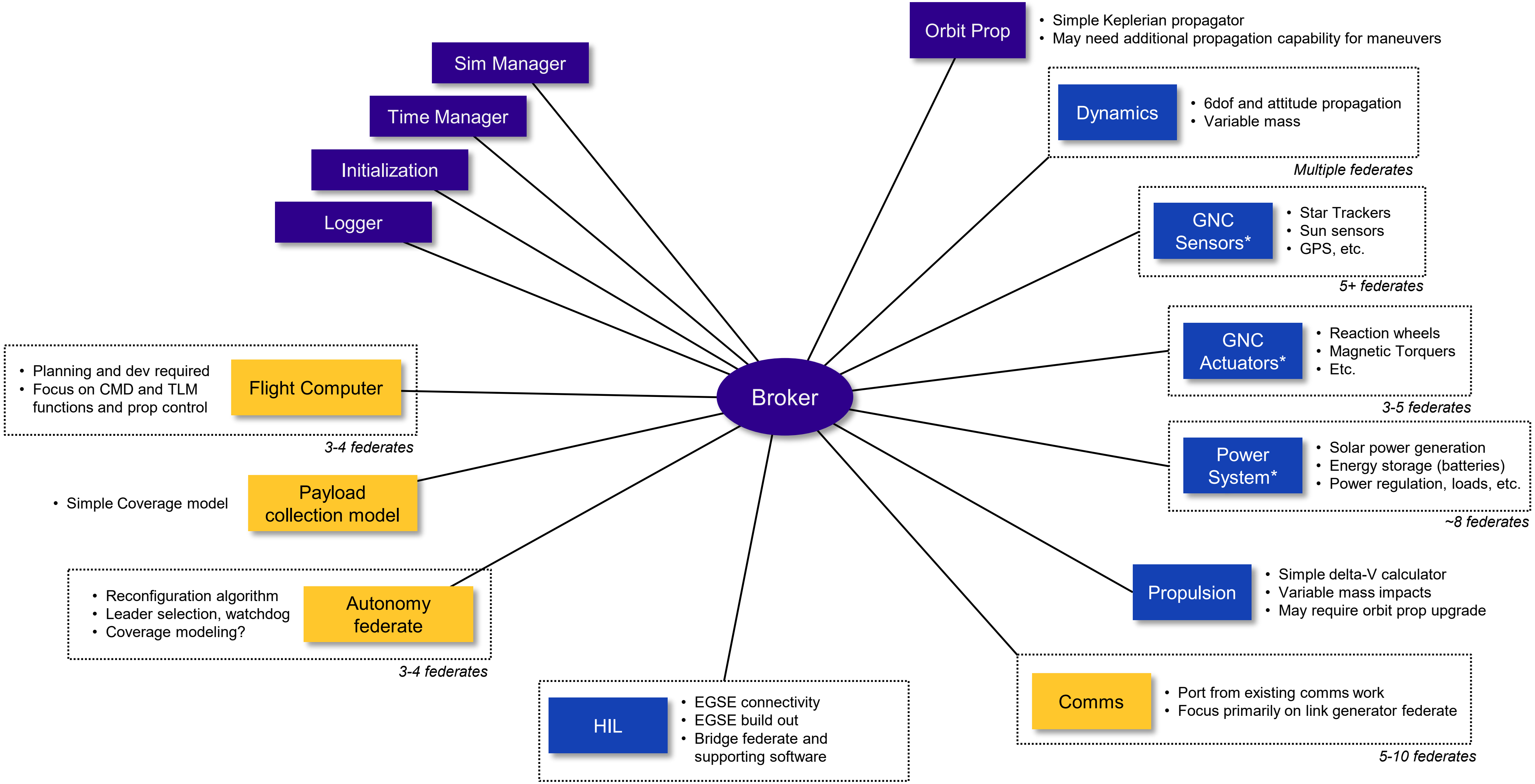
Fault

- Aspen leverages the existing Amigos federated modeling and simulation framework
- Uses a central broker with known IP address which Federates connect to
  - *Federates are unaware of other Federates, Federates can drop in and out without necessarily impacting other Federates*
- Federates ***subscribe*** to data they need and ***publish*** their results (pub/sub)
- Fast and flexible asynchronous message-driven architecture

Federates wrap simulations, emulations, and hardware, broker handles data messaging between federates

Aspen Federation

Currently in development and planned



Broker

Sim Manager

Time Manager

Initialization

Logger

Orbit Prop
- Simple Keplerian propagator
- May need additional propagation capability for maneuvers

Dynamics
- 6dof and attitude propagation
- Variable mass

Multiple federates

GNC Sensors*
- Star Trackers
- Sun sensors
- GPS, etc.

5+ federates

GNC Actuators*
- Reaction wheels
- Magnetic Torquers
- Etc.

3-5 federates

Power System*
- Solar power generation
- Energy storage (batteries)
- Power regulation, loads, etc.

~8 federates

Propulsion
- Simple delta-V calculator
- Variable mass impacts
- May require orbit prop upgrade

Comms
- Port from existing comms work
- Focus primarily on link generator federate

5-10 federates

HIL
- EGSE connectivity
- EGSE build out
- Bridge federate and supporting software

Autonomy federate
- Reconfiguration algorithm
- Leader selection, watchdog
- Coverage modeling?

3-4 federates

Payload collection model
- Simple Coverage model

Flight Computer
- Planning and dev required
- Focus on CMD and TLM functions and prop control

3-4 federates

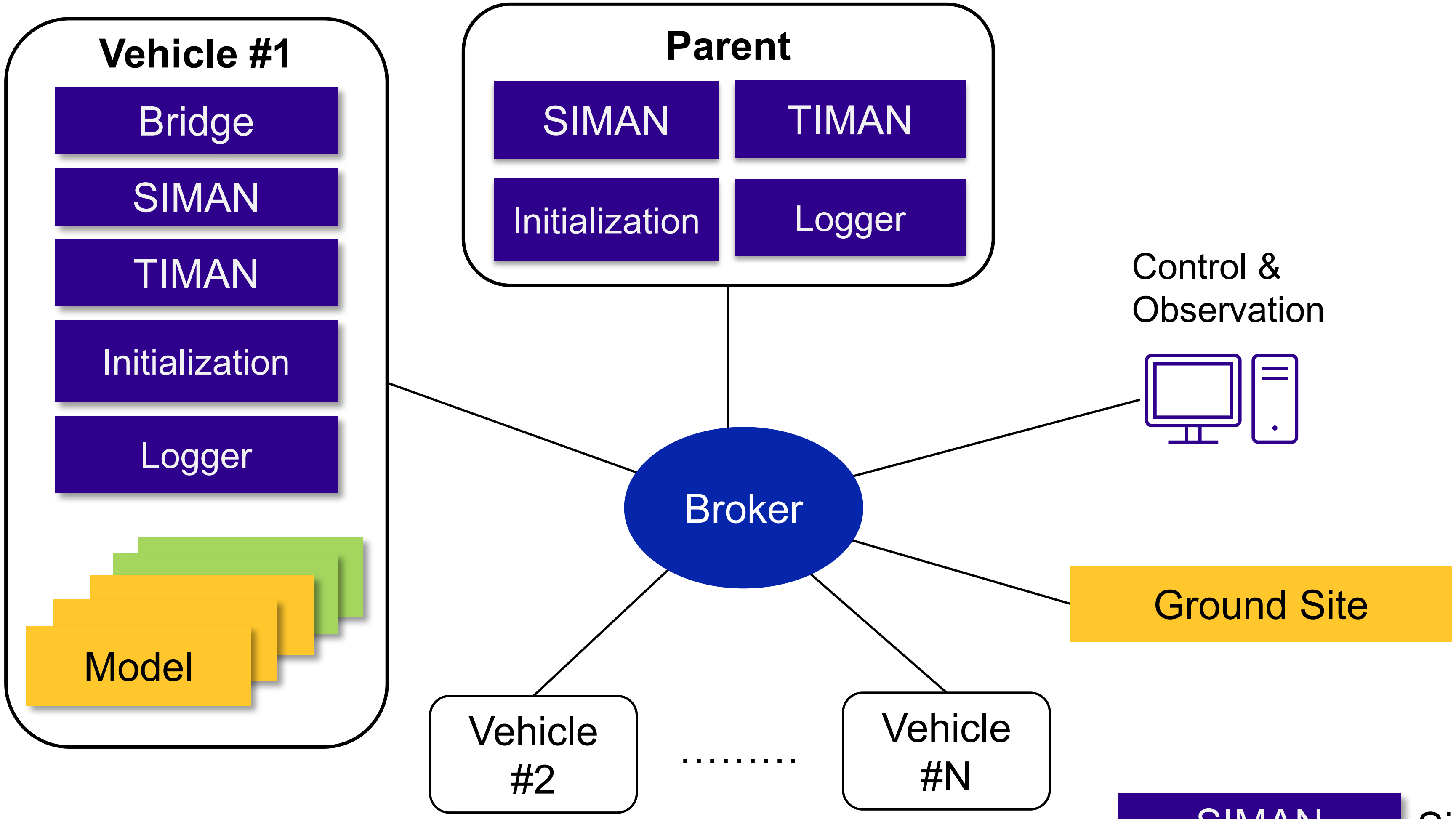
Existing

In Development*

Future Work

Multiple Vehicle Architecture

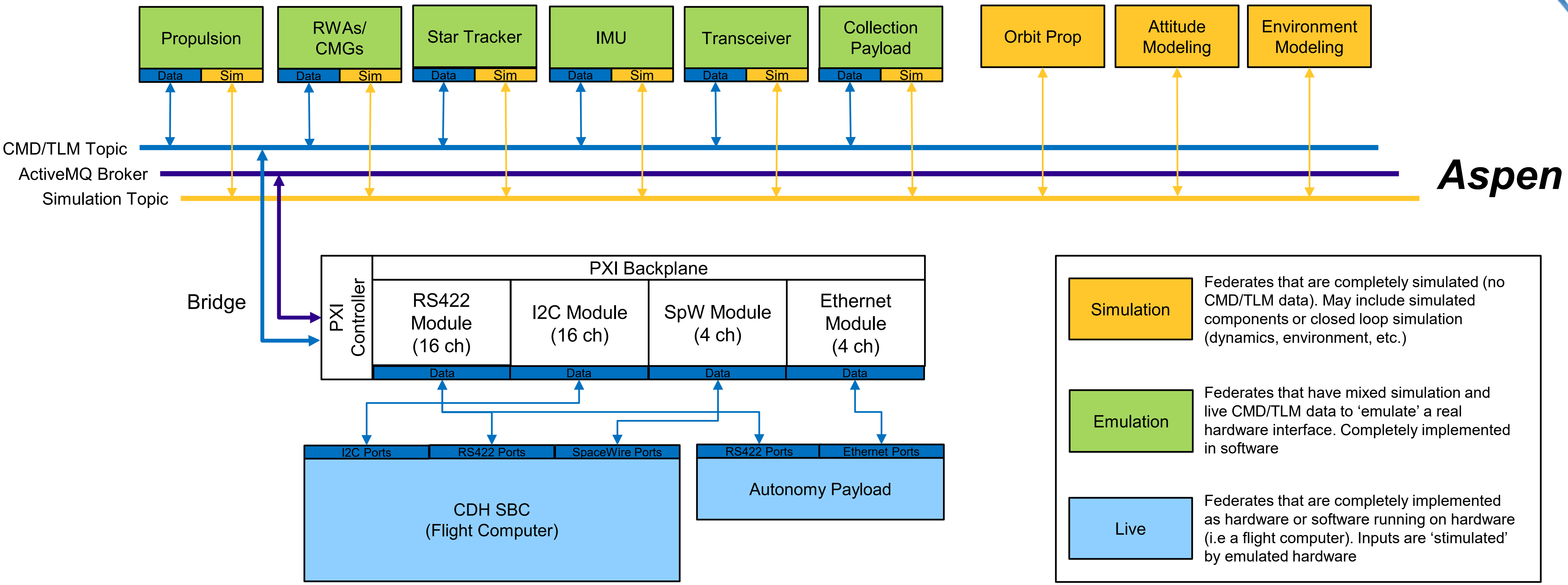
Parent and Child Federations



SIMAN	Simulation Manager
TIMAN	Time Manager

Allows construction of systems of systems

Aspen HIL Implementation



Propulsion
Data | Sim

RWAs/ CMGs
Data | Sim

Star Tracker
Data | Sim

IMU
Data | Sim

Transceiver
Data | Sim

Collection Payload
Data | Sim

Orbit Prop

Attitude Modeling

Environment Modeling

CMD/TLM Topic

ActiveMQ Broker

Simulation Topic

Aspen

Bridge

PXI Controller

PXI Backplane

RS422 Module (16 ch)
Data

I2C Module (16 ch)
Data

SpW Module (4 ch)
Data

Ethernet Module (4 ch)
Data

I2C Ports | RS422 Ports | SpaceWire Ports

CDH SBC (Flight Computer)

RS422 Ports | Ethernet Ports

Autonomy Payload

Simulation	Federates that are completely simulated (no CMD/TLM data). May include simulated components or closed loop simulation (dynamics, environment, etc.)
Emulation	Federates that have mixed simulation and live CMD/TLM data to ‘emulate’ a real hardware interface. Completely implemented in software
Live	Federates that are completely implemented as hardware or software running on hardware (i.e a flight computer). Inputs are ‘stimulated’ by emulated hardware

- Simulation federates produce necessary data to represent the space environment and dynamics/physics of the space vehicle while testing is occurring on earth
- Emulation federates represent individual satellite components in place of real hardware, creating stimulus telemetry for real hardware and software in the loop
- Real hardware is connected to the federation through a bridge to a National Instrument PXI flexible interconnect system to provide various forms of electrical connectivity to the hardware. Data from emulation federates are streamed over these physical channels

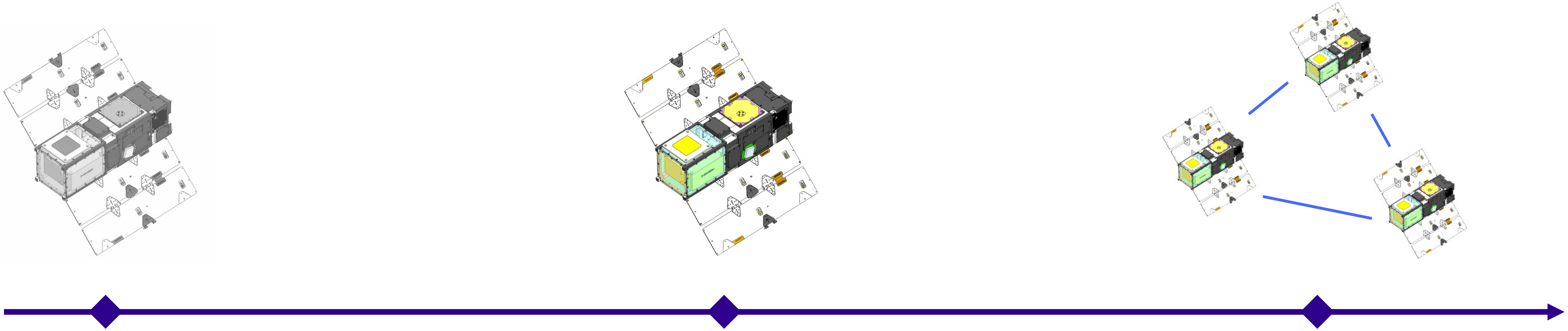
Aspen Hardware Interfacing

- Relies on NI PXI for Electronic Ground Support Equipment (EGSE)
- Amigos framework acts as the connective layer between federates
- PXI acts as the translation layer between software and hardware
  - *Aspen connects directly to hardware through the PXI*
  - *Aspen development team working on abstracting away all of the required interface development with the PXI*
  - *PXI involves a plethora of various software including C APIs, FPGA coding, and firmware to function properly*
- End goal for Aspen is to develop a turn-key bridge to hardware through the PXI so that a user can choose a serial port on the PXI and plug in their device to the necessary ports
  - *New PXI modules, serial interface protocols, etc. will always necessitate additional development and NRE*
  - *Over time, Aspen team will build up a library of available interfaces to have available for customers wanting to run HIL tests*
- PXI was decided upon because it is:
  - *Industry standard EGSE, allowing our Aerospace team to use and get comfortable with the same hardware the contractors we evaluate use*
  - *PXI has standardized implementation of a wide variety of modules allowing for almost unlimited customizability for hardware connectivity, it is the gold standard*

PXI translates between Aspen simulations and live hardware under test

NI PXI unit setup in VED Vehicle Autonomy Laboratory in Chantilly, VA

Aspen Long Term Strategy



End of FY22
- Framework developed
- Zero/First order simulations (bus systems)
- Hardware interfacing functional
- Lessons learned and deltas on path ahead

FY23
- Framework improvements
- Model fidelity enhancements
- Extension to payload modeling
- Flight software in the loop capability
- Further development of HIL capability (PXI)
- Testing with high performance computing board running flight software/algorithms
- Testing with autonomy agent and development of autonomy testbed
- Digital Twin platform development

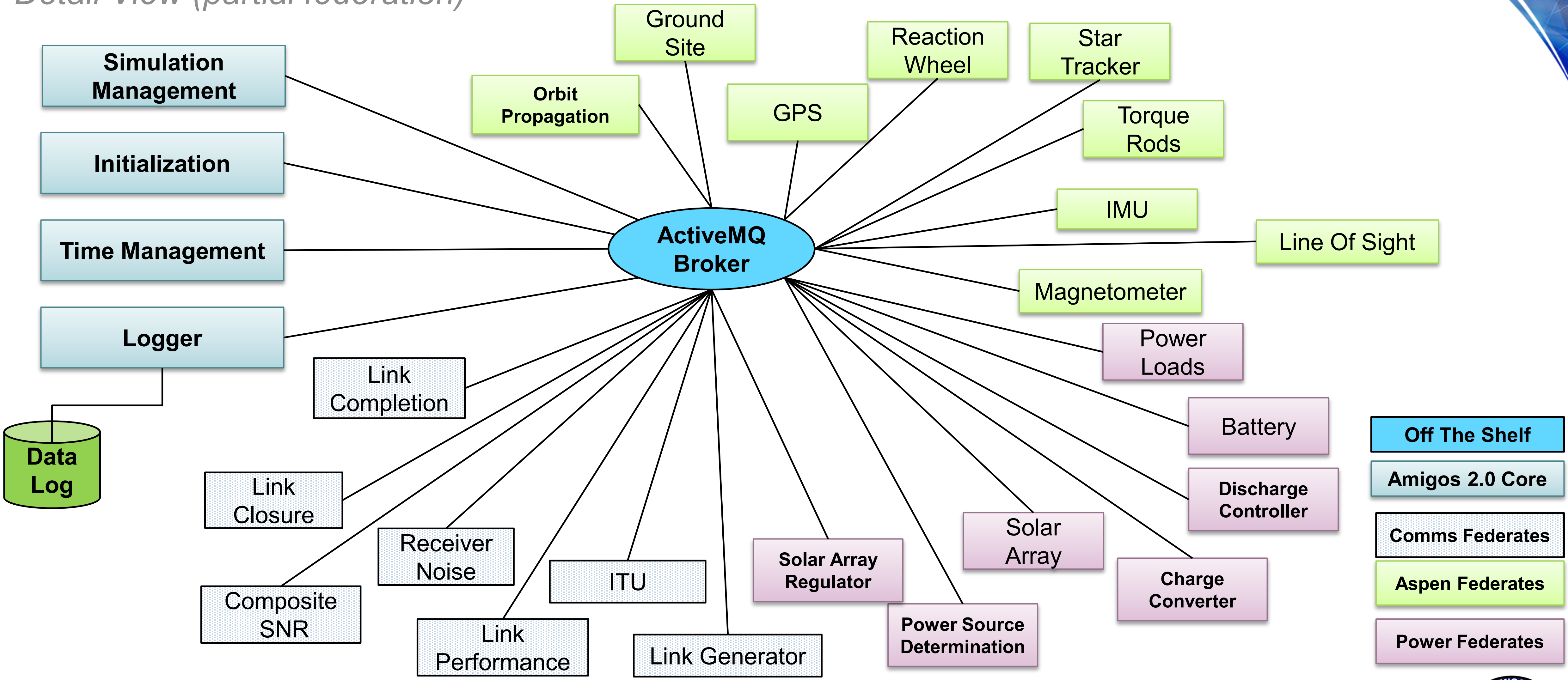
Future (FY24+)
- Multi-satellite operations – RPO, formation flight, collaborative ops, mesh networking
- Multi-satellite autonomy HIL/SIL/MIL testbed
- Robust fault simulation capability for stress testing and V&V of autonomous systems
- Functioning digital twin platform for mission modeling

Developing HIL/SIL/MIL capability for multi-satellite autonomous systems

Backup

Aspen Federation

Detail View (partial federation)



Aspen federation would support many individual federates and capabilities