34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

Using VR to Validate &Visualize MBSE-Designed Interfaces

Céad Míle Fáilte (Welcome)

Sean Flanagan

- Senior Model Based Systems Engineer at **i³**
- Software Systems Architect
- Experience with Uncrewed Aircraft Systems and Digital Engineering initiatives
- B.S. in Computer Science at Oklahoma State University
- M.S. in Software Engineering at Southern Methodist University

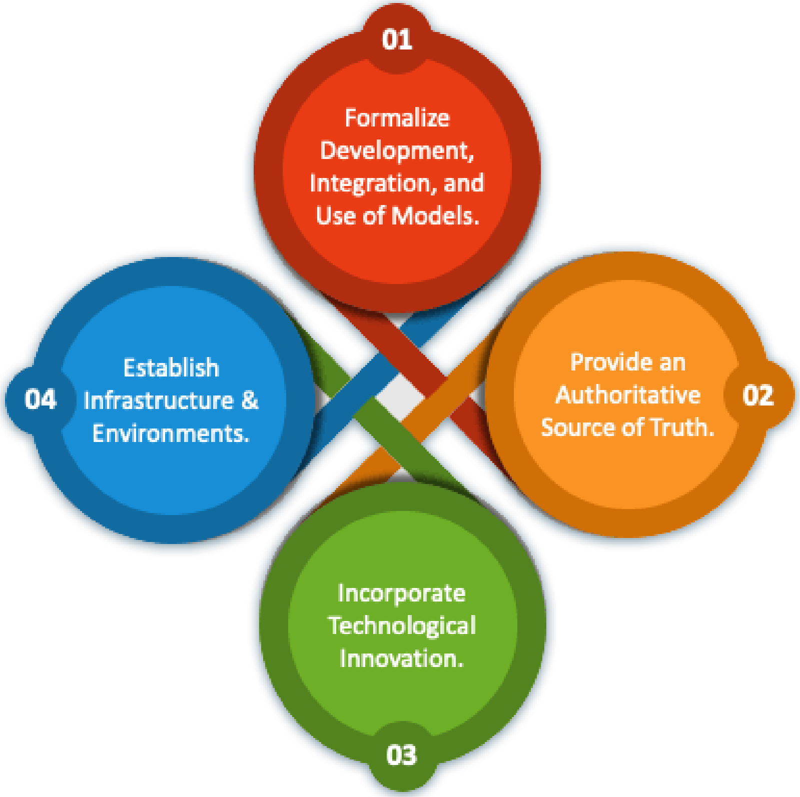
Introduction

- 2018: DoD publishes their strategy for the adoption of **digital engineering (DE)** into the nominal acquisition process

- **Model-based systems engineering (MBSE)**: one of the several legs in this DE strategy, on which industry and government are both working to make the transition

- Critical to **MBSE's success**: a shift in process and execution that encourages an **organizational culture** around it

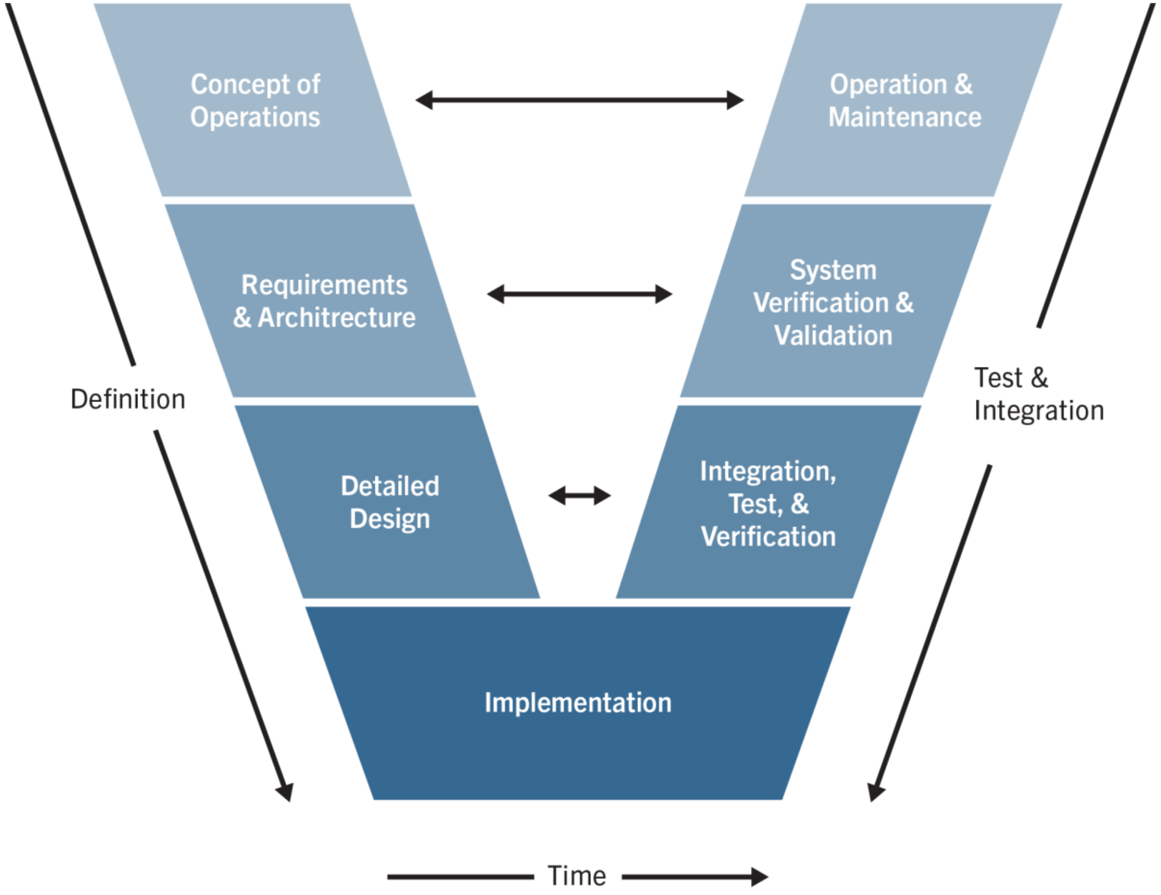
- **Our focus:** continually seeking **innovative approaches** as well as subject matter expertise to help develop and execute these strategies


01
Formalize Development, Integration, and Use of Models.
02
Provide an Authoritative Source of Truth.
03
Incorporate Technological Innovation.
04
Establish Infrastructure & Environments.

DoD Digital Transformation Strategy, 2018; McDermott, et al., 2020

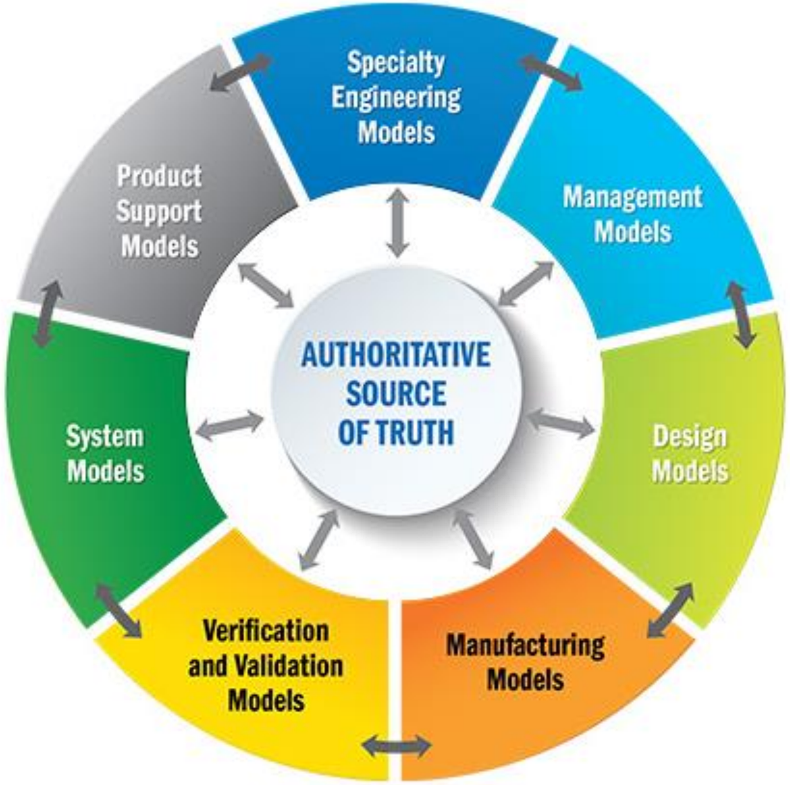
The Perception

- **Perceived barriers to MBSE adoption:**
  - Excessive **time** spent in the architecture & design phase *v.* traditional approaches
  - **Cost / effort budget** will expend prior to reaching practical tests (SWILSIM, HILSIM, integration, etc.)
  - Expectation that spending more time in the design phase assures an end-product is "**trouble free**" when built
  - The longer the MBSE design goes without contact with the real world, the greater the **risk** that is carried when the system finally goes to test



The Perception

- **Perceived concerns with tooling:**
  - Difficulty in identifying and adopting the right MBSE solution as innovation brings about new tools
  - Concern of being ‘locked’ into a particular solution, resulting in incompatibility with other toolsets
  - Fear that existing in-house simulations & analysis tools, many with a degree of high-fidelity, are not ‘MBSE-ready’ or compatible with existing MBSE tools


Specialty Engineering Models
Management Models
Design Models
Manufacturing Models
Verification and Validation Models
System Models
Product Support Models
AUTHORITATIVE SOURCE OF TRUTH

The Problem

Is there a way to “buy back” integration and test time by doing verification level testing with the model?

Is there a way to link toolsets from different ecosystems?

The Premise

Switch Examples
Dial Examp

Point I

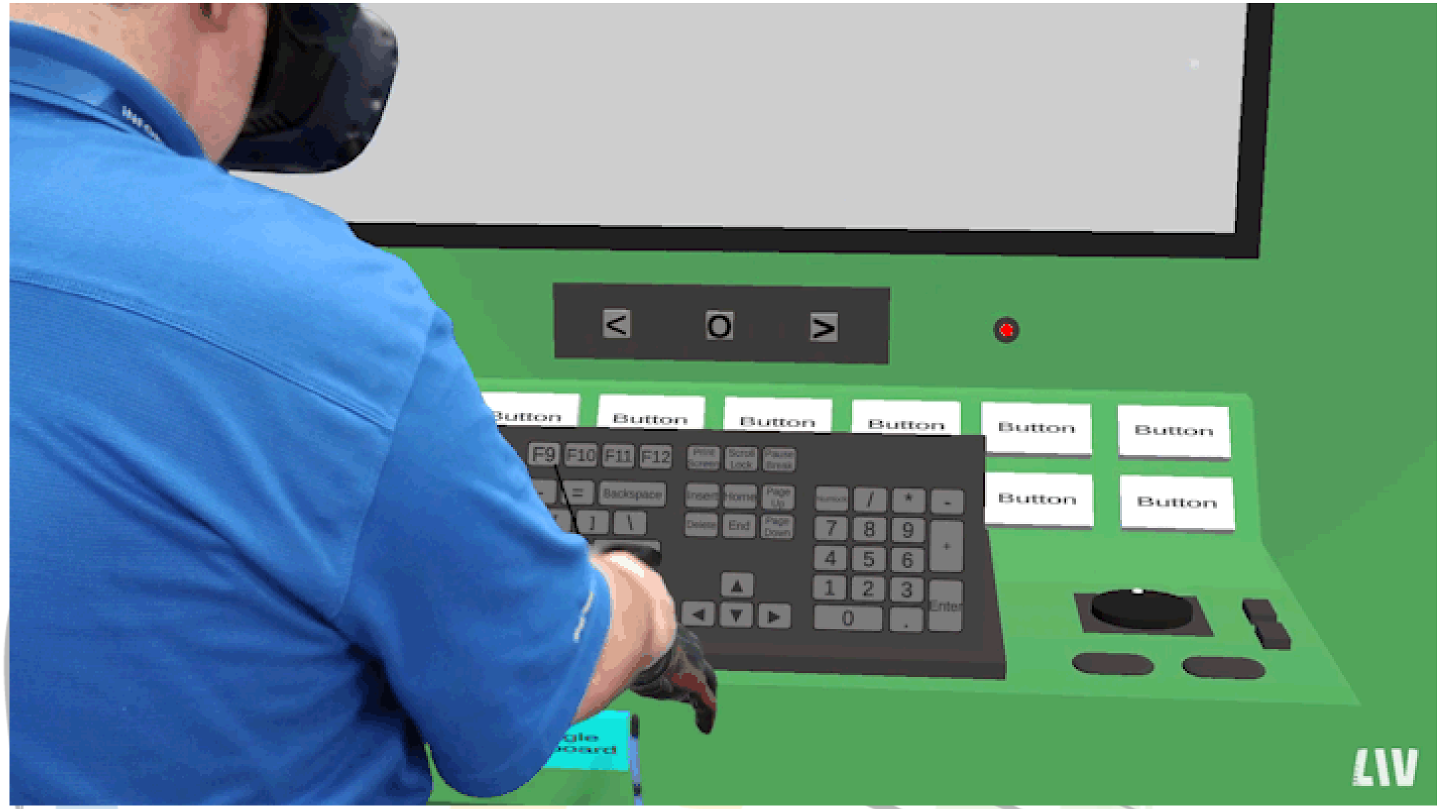
Our study explored the idea that a **Virtual Reality system** could serve as a surrogate test environment, a kind of “virtual HILSIM”, which could help drive MBSE architectures towards a more relevant solution.

The Premise

Point II

By using a virtual, not physical, form factor, it allowed for the physical aspects of the system to be **modified at the speed of MBSE iterations** and, conversely, provided **real-time feedback** to the virtual user interface.

The Premise



Point III

This virtual presence served as a “half step” towards a HILSIM test – a **“HILSIM in a headset” that helped to incrementally increase the realism** of the test environment in a useful way, all while **validating the interfaces and logic** of the modeled design.

The Approach

BIFROST

The Bi-Directional Interface Requirements Operational System Tool (BIFROST)
Bridging the gap between an executable system model and the outside world

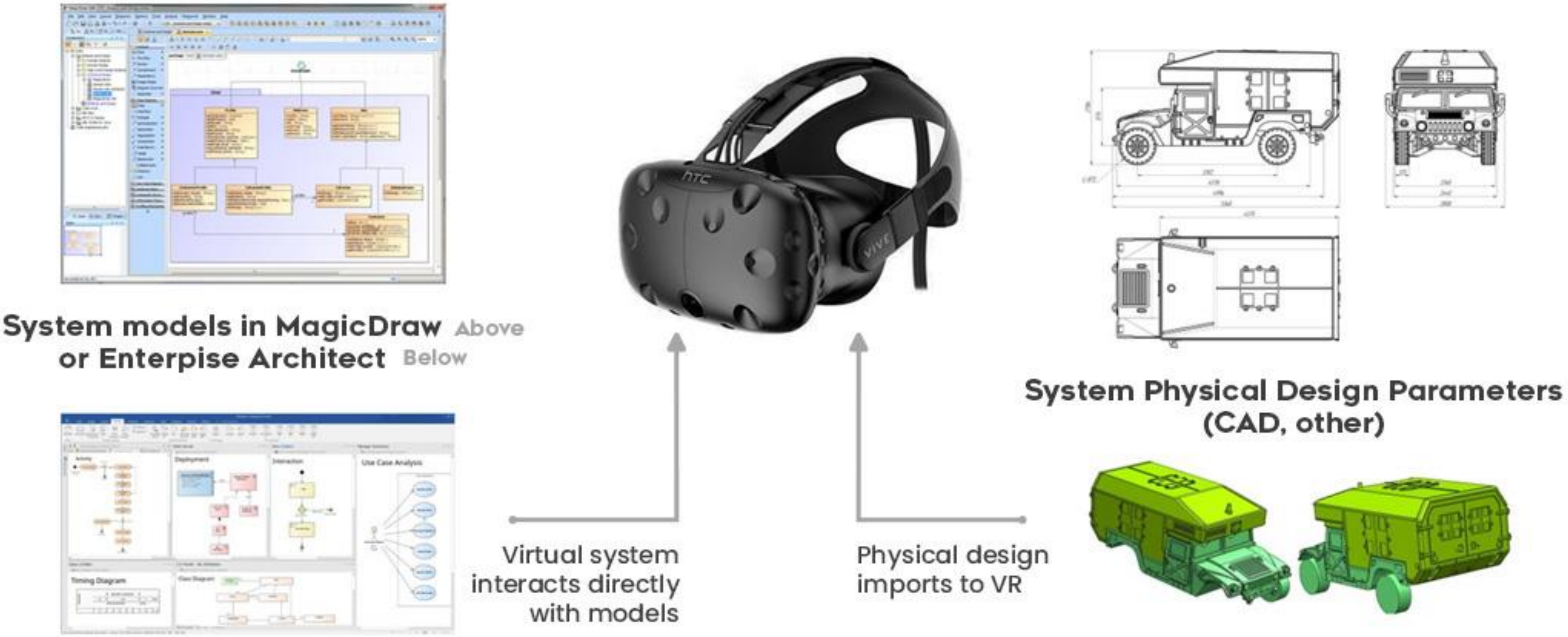
Three (3) distinct tests:

1 BLACK BOX TESTING

2 WHITE BOX TESTING

3 USER INTERFACE TESTING

The Approach



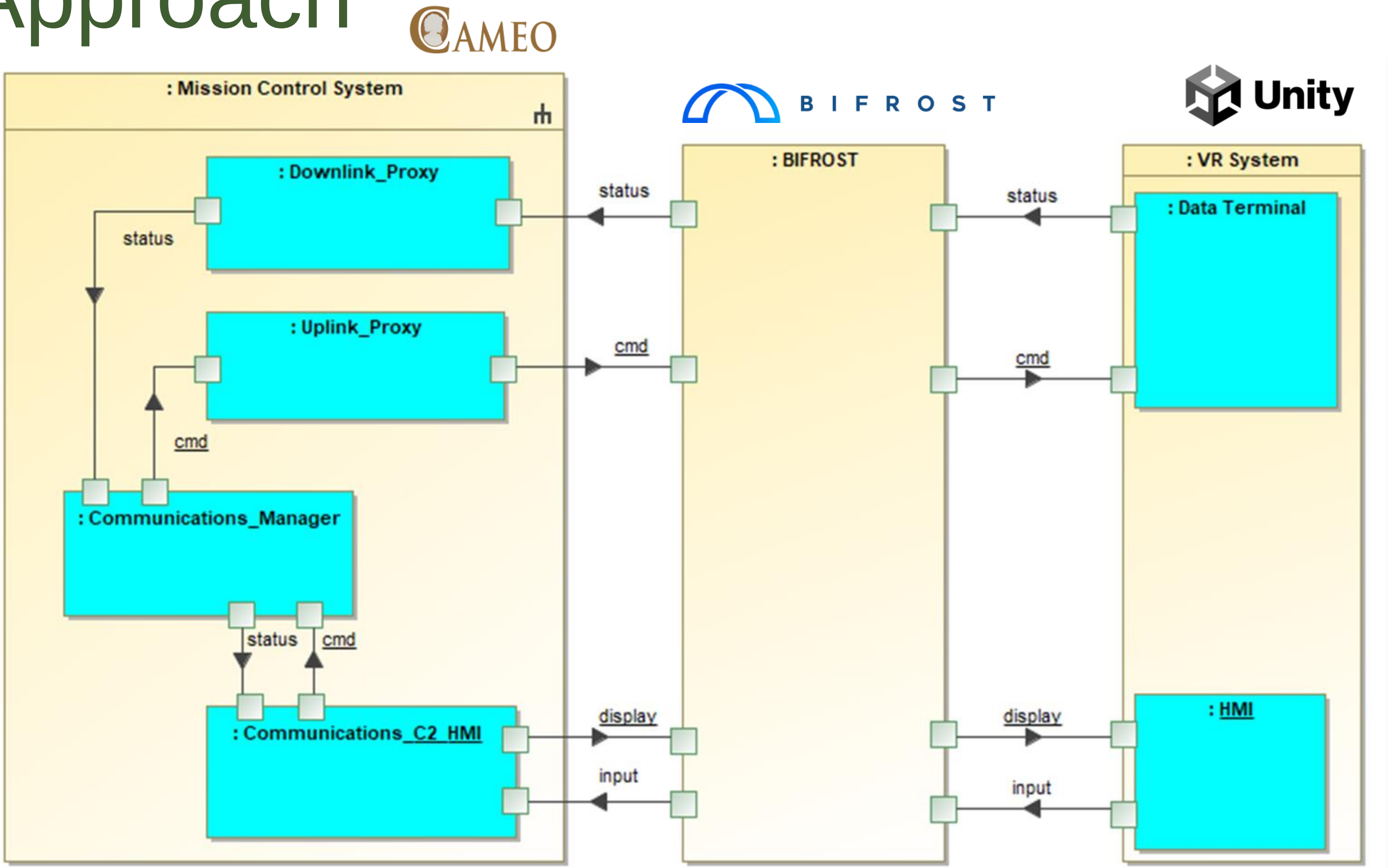
System models in MagicDraw Above
or Enterpise Architect Below

System Physical Design Parameters
(CAD, other)

Virtual system
interacts directly
with models

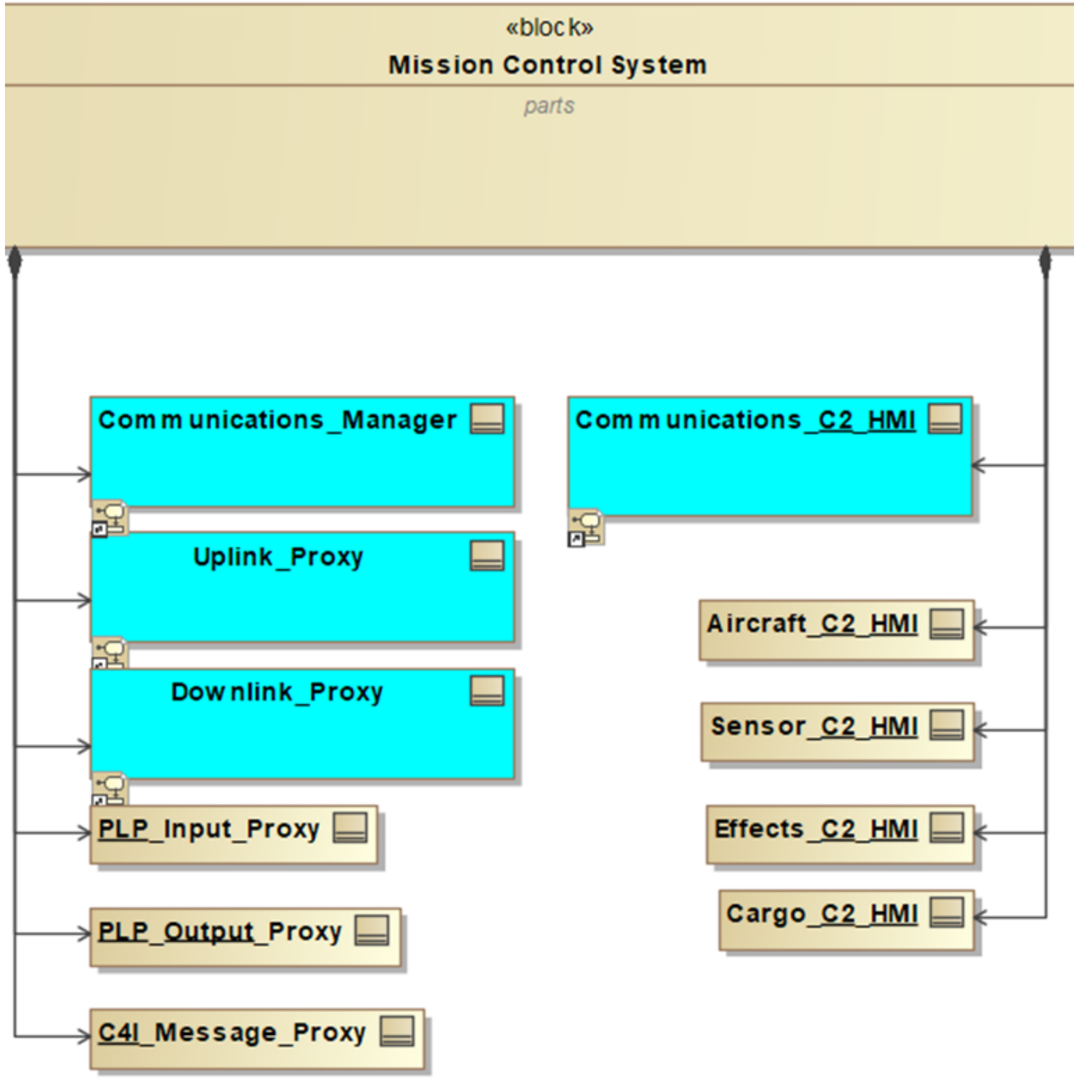
Physical design
imports to VR

The Approach



The Implementation

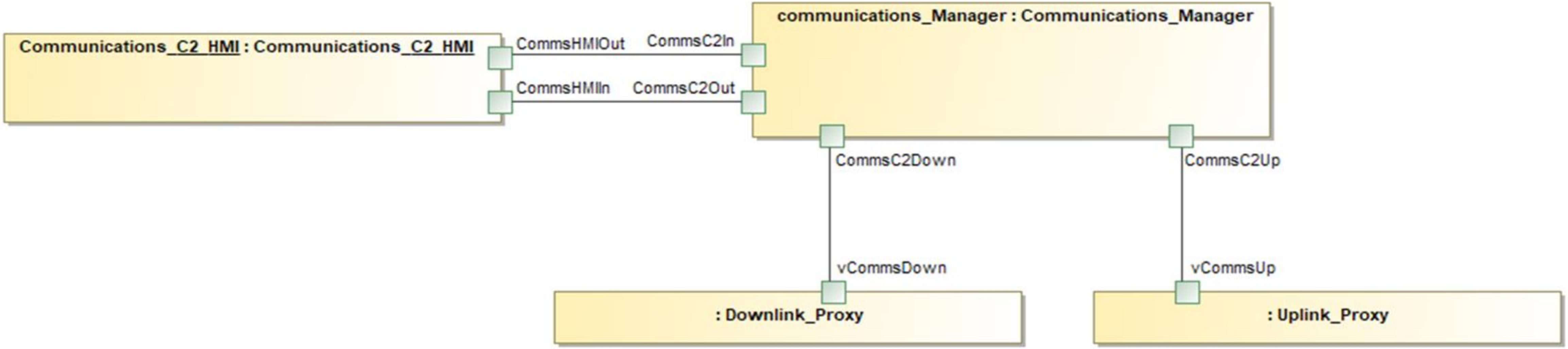
- The starting point for the study was an existing Mission Control System Architecture model that had been developed. The logical structure, behavior, and interfaces were fully modeled.
- The study was scoped to incorporate only the use cases pertaining to data terminal control.



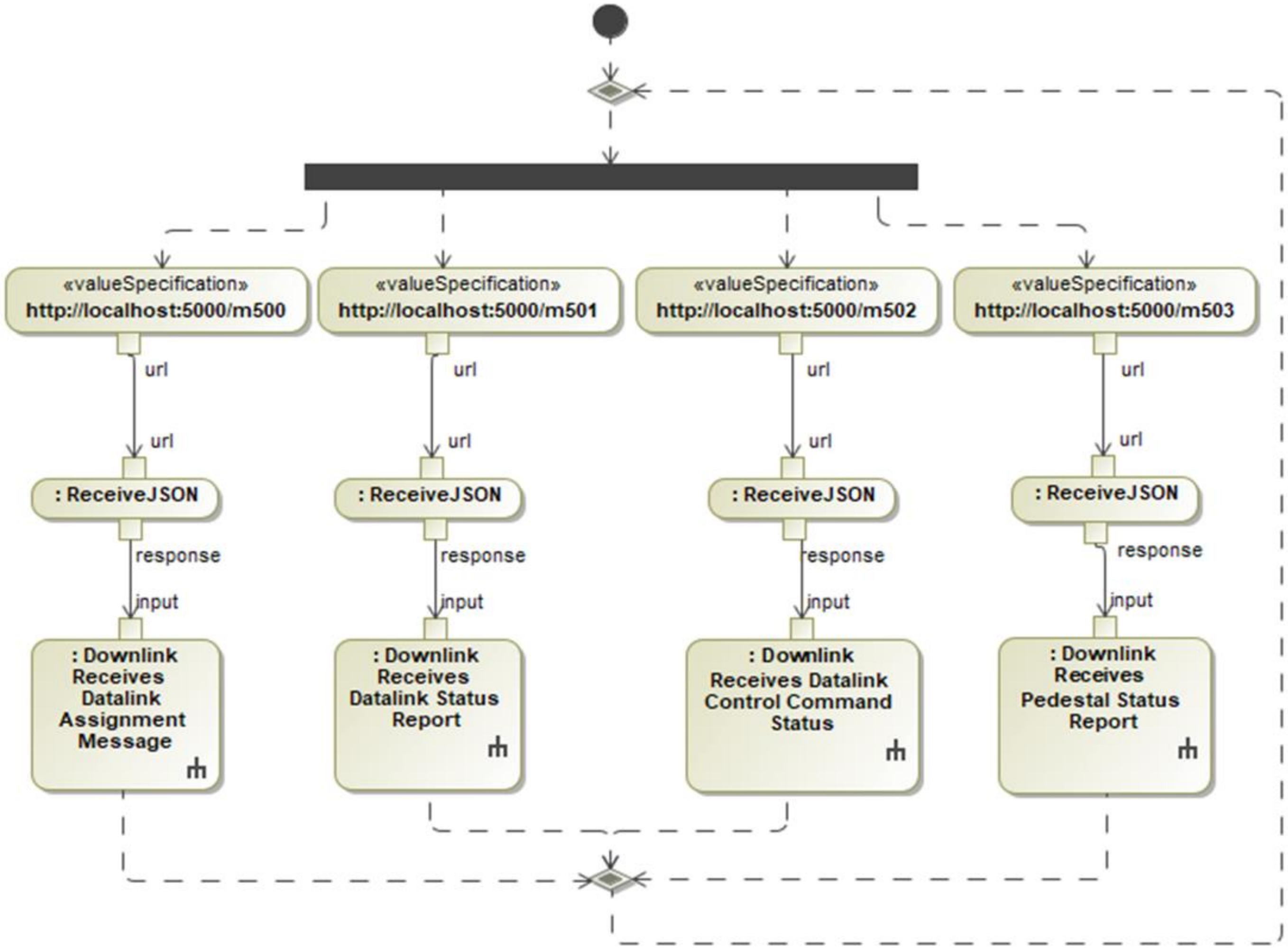
The Implementation

- The model was extended to make it “execution ready” by creating additional activity and internal block diagrams to parallel existing content.

- This allowed us to test our concept without altering the original design.



The Implementation

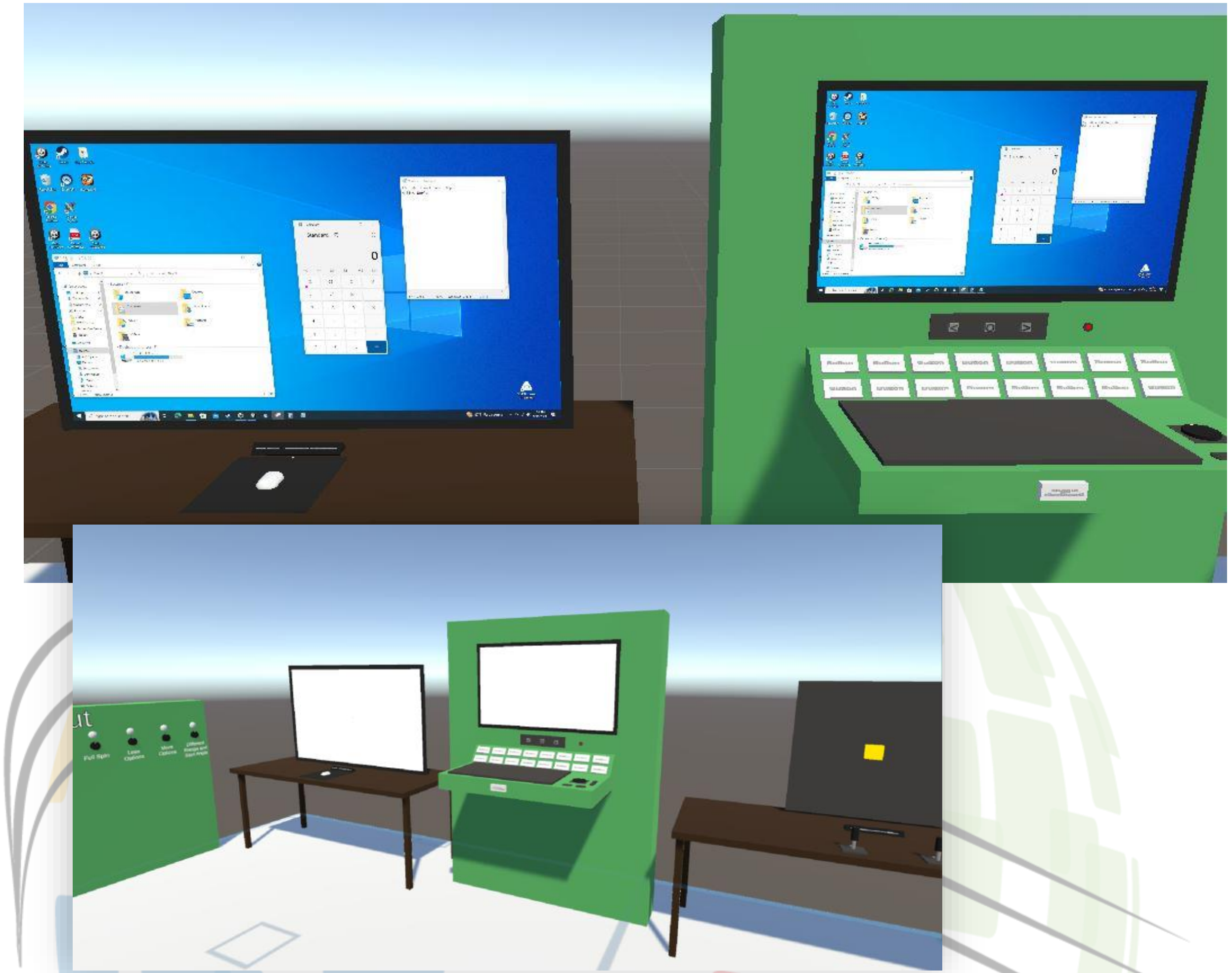

«valueSpecification»
http://localhost:5000/m500
«valueSpecification»
http://localhost:5000/m501
«valueSpecification»
http://localhost:5000/m502
«valueSpecification»
http://localhost:5000/m503
url
url
: ReceiveJSON
response
input
: Downlink Receives Datalink Assignment Message
: Downlink Receives Datalink Status Report
: Downlink Receives Datalink Control Command Status
: Downlink Receives Pedestal Status Report

- Opaque actions were inserted into the activity diagrams to handle the sending/receiving of messages from BIFROST.

- Opaque action code was written in JavaScript.

- Messages were converted to/from JSON objects and transmitted to BIFROST via REST API calls.

Visualization Environment



- Virtual Reality:

Many benefits to **validating UX for performing a task**

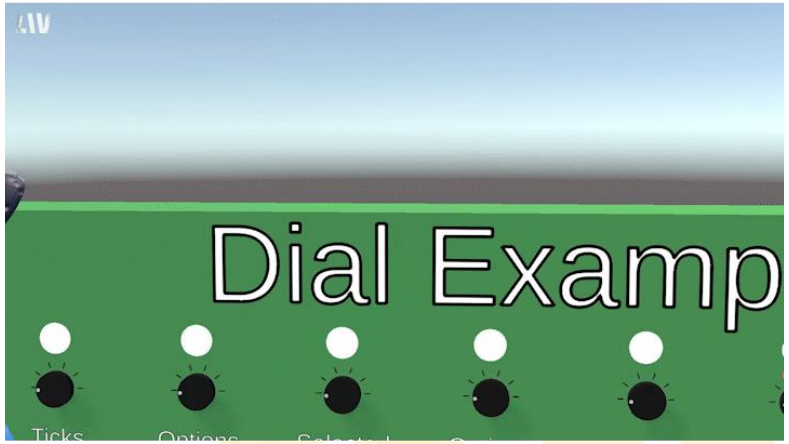
- **Networked** systems:

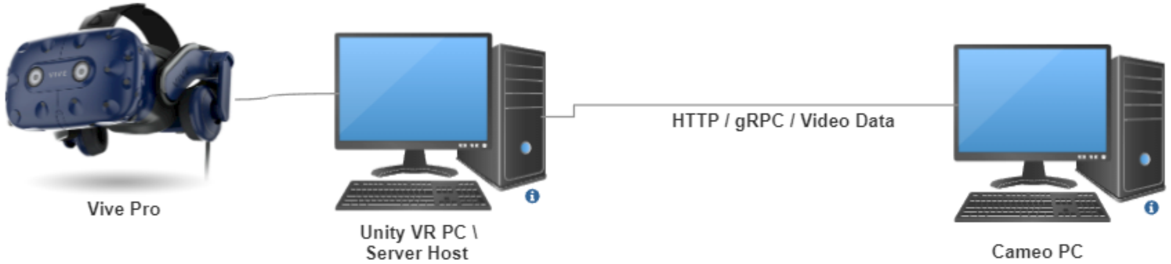
Virtual environment communicates with the system's model in **a bi-directional format**

- **Shared** screens between systems
- Simulated **physical controls** along with simulated or actual **GUIs**
- Interaction with other connected systems in near **real-time**

All 3D models were made for function and not representative of final product's visuals

Visualization Environment





All 3D models were made for function and not representative of final product's visuals

Lessons Learned & Future Research

LESSONS LEARNED

- While adapting an existing model is possible, it is much easier to build a model for execution at the onset.
- Due to system performance, it is preferable to run the executable model on its own platform separate from the second system.
- While JSON is convenient, there are better formats available for handling complex data types and enumerations.
- Integrate model and virtual environment early.

FUTURE RESEARCH

- Investigate how using this methodology as a **systems trainer** can reduce expected training time and evaluate new use cases.
- **AI/ML** – We have successfully communicated with ChatGPT from a model, sending queries and capturing responses.
- **Mathematical Models** – Pass inputs into a model and capture data to use in parametric analyses.
- **Simulations** – Communicate directly with simulations to perform automated test.

34th Annual INCOSE international symposium

hybrid event

Dublin, Ireland
July 2 - 6, 2024

www.incose.org/symp2024
#INCOSEIS

Go raibh maith agat (Thank You)