ITEC

VT MAK

A company of VT Systems

The Trend Toward Common Architectures

Pete Swan, Director International Sales

The Vision

"These efforts are focused on creating a synthetic environment that will support thousands of entities spread across many sites and provide dynamic terrain, weather, phenomena, simulated command forces, and more complex terrain. The architecture must:

- Scale from a few simulations on a LAN through a large simulation with 100,000 entities from 50 sites.
- Provide a real-time system.
- Support Live, Constructive, and Virtual simulations.
- Support environmental phenomena.
- Support changing network technology and network topologies.
- Reach small sites (possibly mobile) linked over low bandwidth lines."

- 11th DIS Workshop 1994

AGENTS: An Architectural Construct to Support Distributed Simulation
James O. Calvin, MIT Lincoln Laboratory
Daniel J. Van Hook, MIT Lincoln Laboratory

Synthetic Training Environment

Common Synthetic Environment

What is STE?

US Army's next-generation vision for a common Synthetic Training Environment for all collective training applications:

- STE is a collective training environment that optimizes human performance within a multi-echelon mixed-reality environment.
- It provides immersive and intuitive capabilities to keep pace with a changing operational environment and enable Army training on joint combined arms operations
- The STE moves the Army away from facility-based training, and instead, allows the Army to train at the point of need — whether at home-station, combat training centers or at deployed locations.

Current State of STE

Currently consists of 3 separate elements

- Software: Common Synthetic Environment (CSE) – **VT MAK**
- Hardware: Reconfigurable Virtual Collective Trainers (RVCT) – **Cole Engineering Services**
- Content: One World Terrain (OWT) – **Vricon**

MAK was awarded the STE CSE contract on June 14, 2019

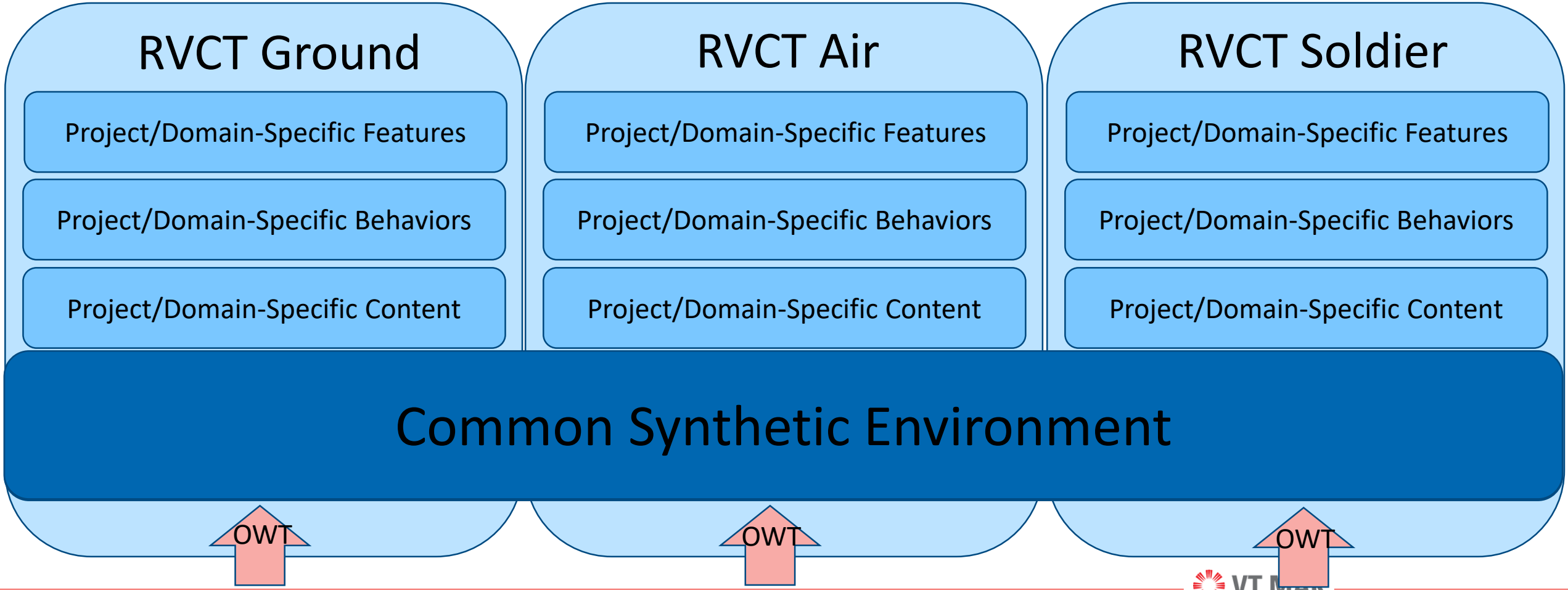
The STE CSE will be based on VR-Forces, VR-Engage, VR-Vantage, VR-TheWorld Server, and other MAK products!

Other limited prototype developments are also being funded

User Assessment 1 – November 2019

The STE Vision

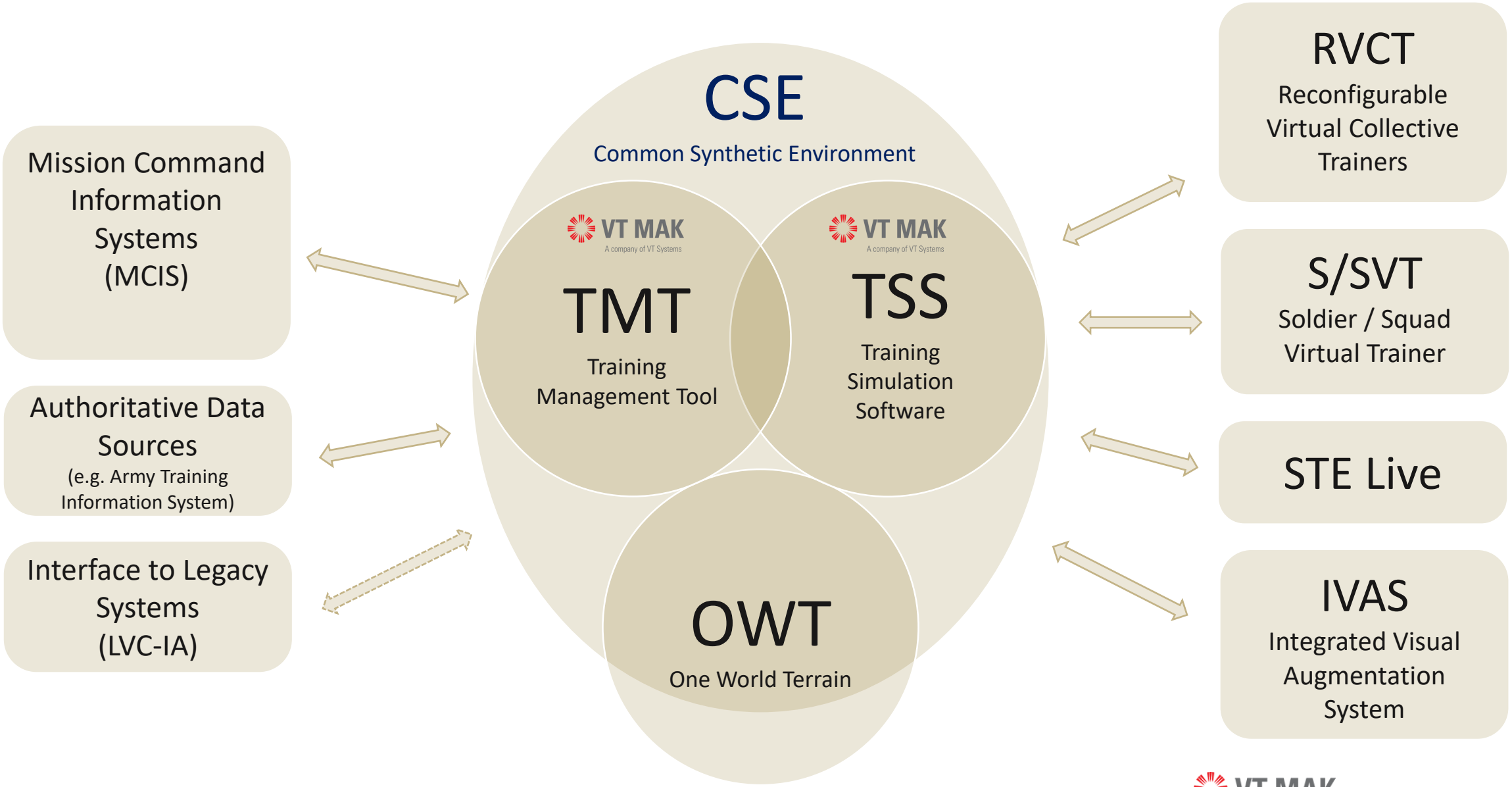
A Common Synthetic Environment and common One World Terrain underpinning the virtual simulators, IG, SAFs across all Collective Trainers



Common Synthetic Environment Characteristics

- Virtual simulators, CGF, and IG built on unified engine
- Solider, ground, air simulators built on unified engine
- Support for whole-earth geo-specific, geo-centric terrain – and ability to load source data
- Modular architecture / open APIs for user extension
- Cloud and Point-of-Need deployment
- Potential for constructive simulation on same engine
- Innovative and agile development process

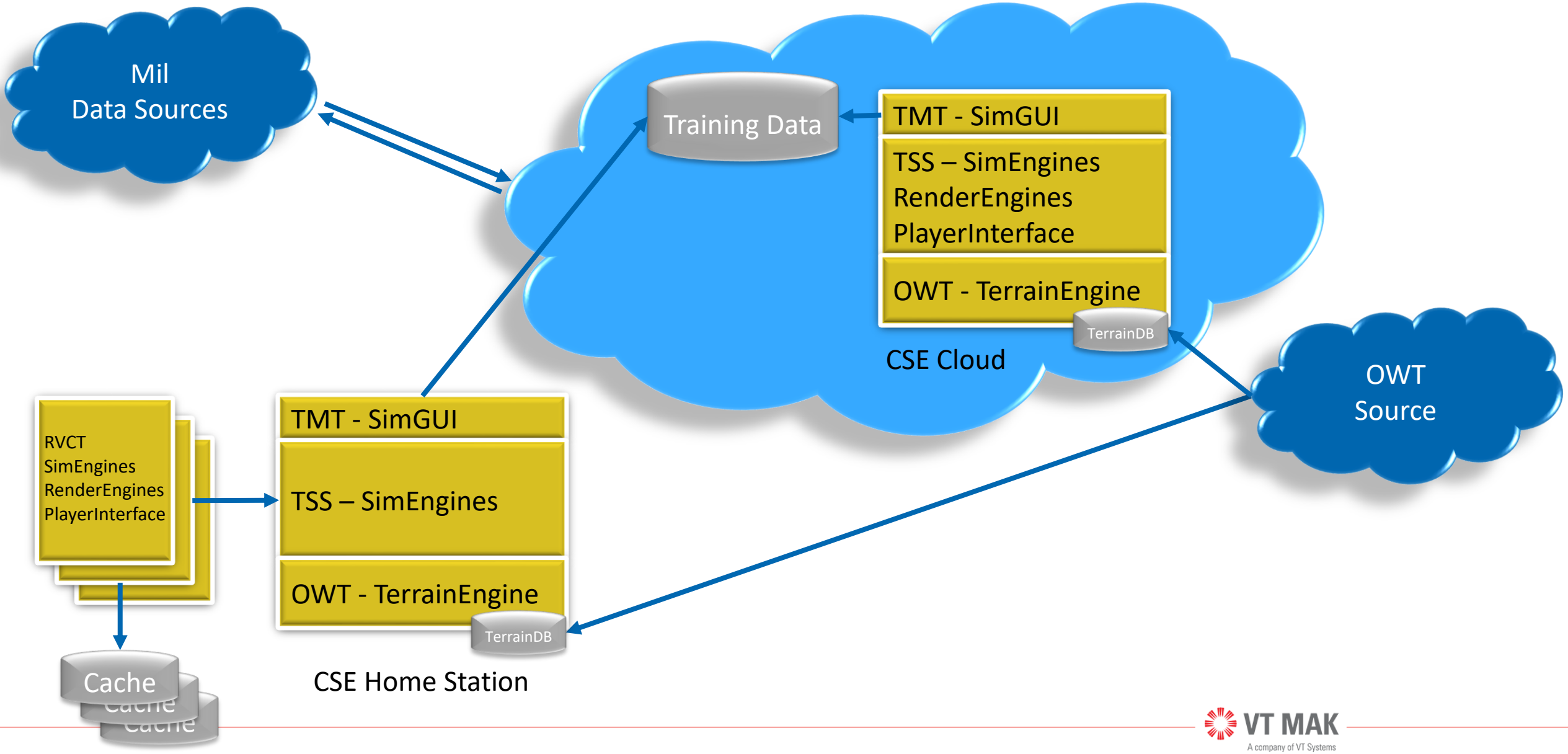
STE Common Synthetic Environment



CSE Architecture

Slide content awaiting release approval.

Hybrid Cloud Model



Cloud Characteristics

Takes advantage of NVIDIA GRID technology

3D graphics rendered on server side, with high-resolution video streamed to client machine

Client does not need to have a high-end graphics card, and can even be a simple virtual desktop infrastructure (VDI)

Supports full HLA federation, using MAK RTI, on virtual subnet on Cloud!

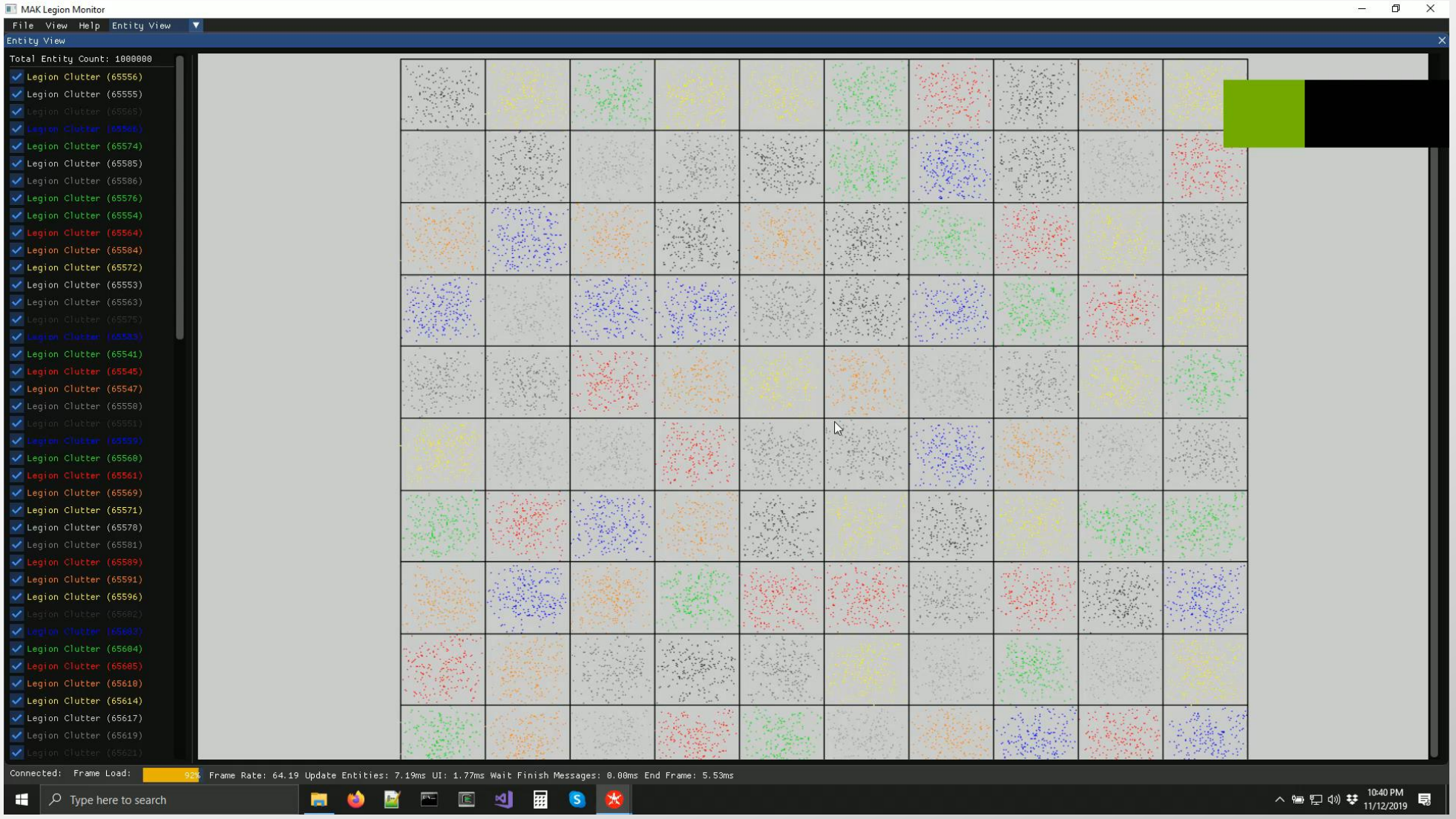
Possible to access EXCON GUI and Role Player/Virtual Simulation from a web-browser with no plug-ins or native applications installed

However, performance and user experience is better with a native application like NICE DCV (relative mouse, etc.)

Mass-market technology continues to advance rapidly, and MAK is continuing to incorporate the latest technologies

Scalability

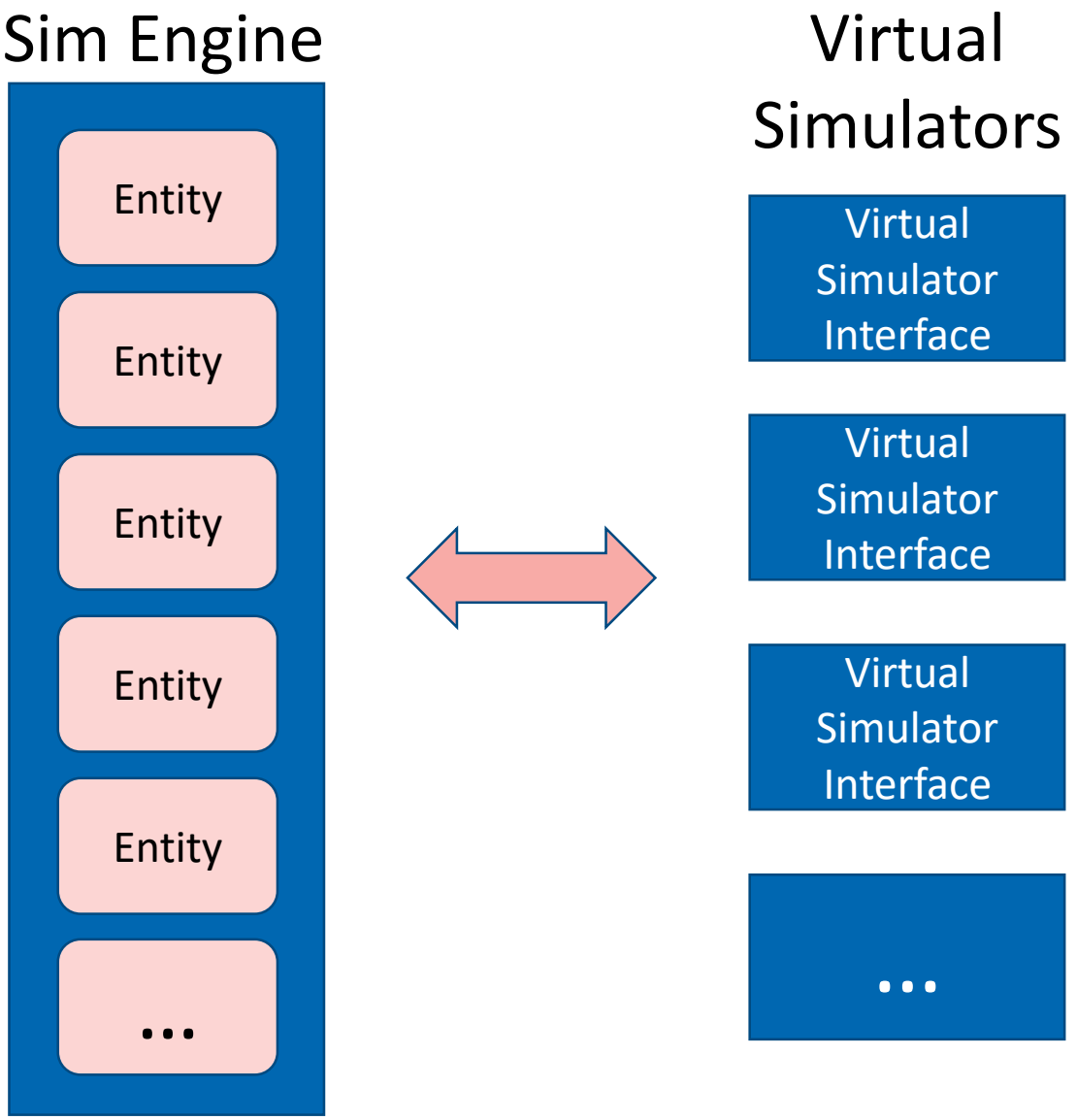
One Million Entities running through MAK's new Legion scalability framework!



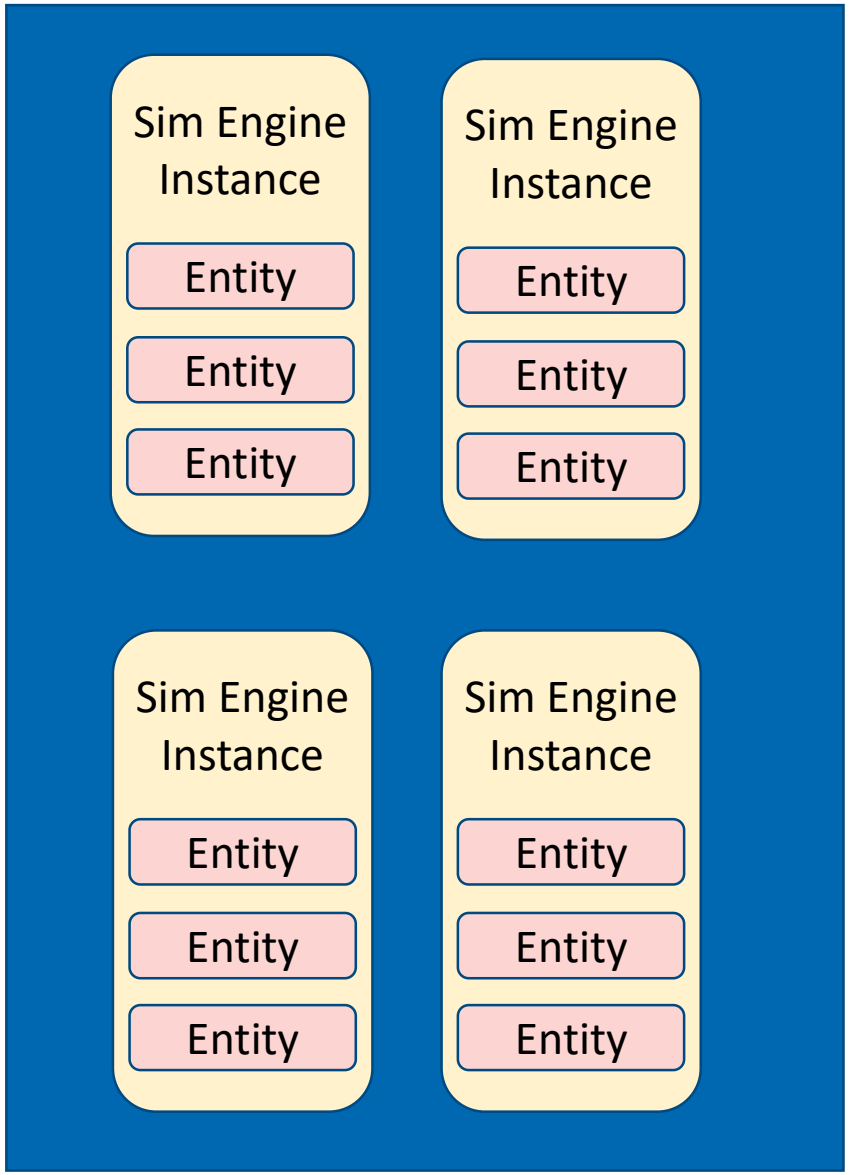
Network Communication Among Sim Engine Instances, and With Virtual Simulators and Other Clients

Top-level Problem

- Sim Engine needs to efficiently simulate large numbers of entities each frame
- Simulation code that computes the state of each entity may require knowledge of state of other entities
- Sim Engine needs to communicate state of entities to Virtual Simulators and other applications
- For player-controlled entities, the code that computes the state of the entities requires input from Virtual Simulator Interface

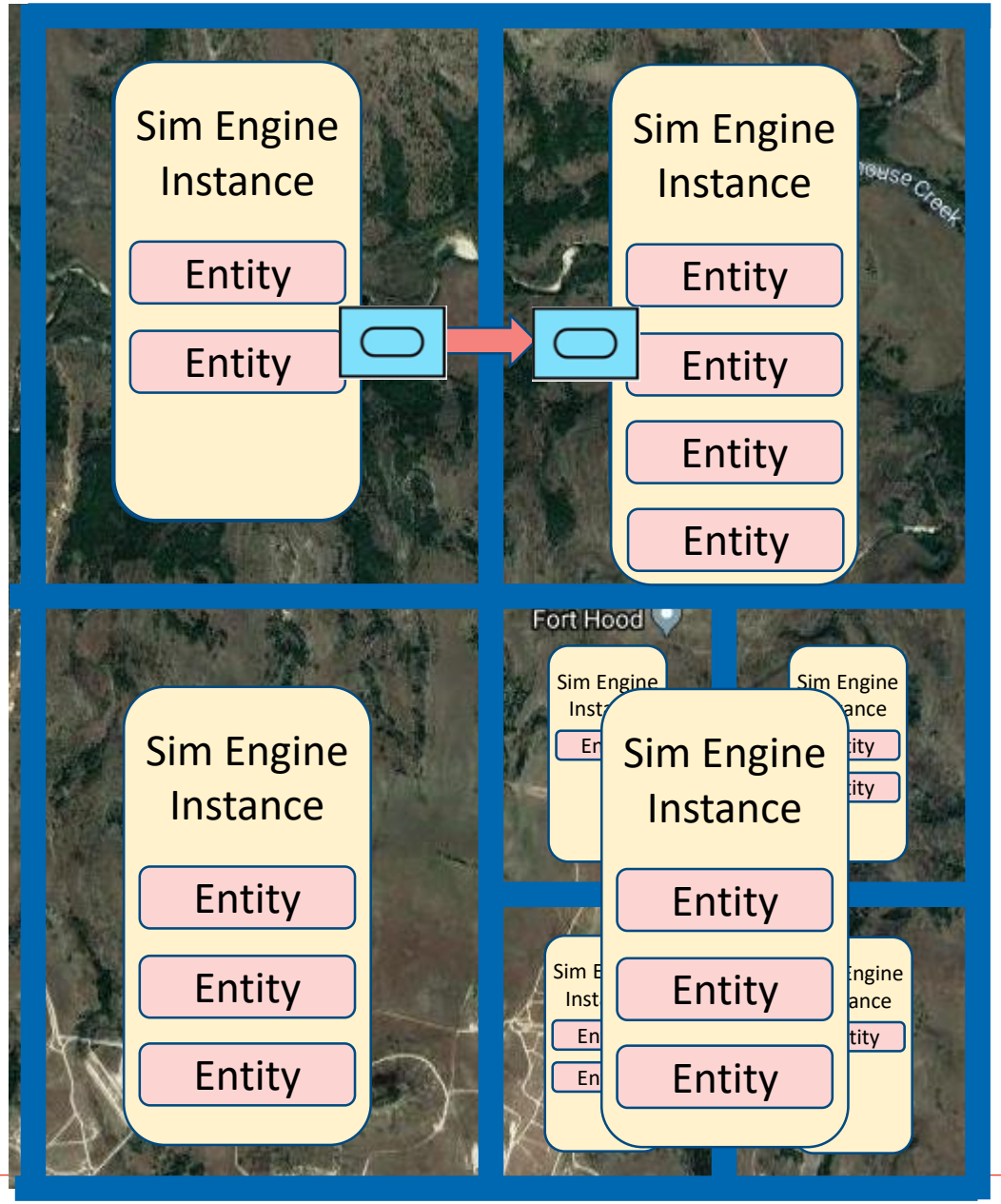


Sharing the Load



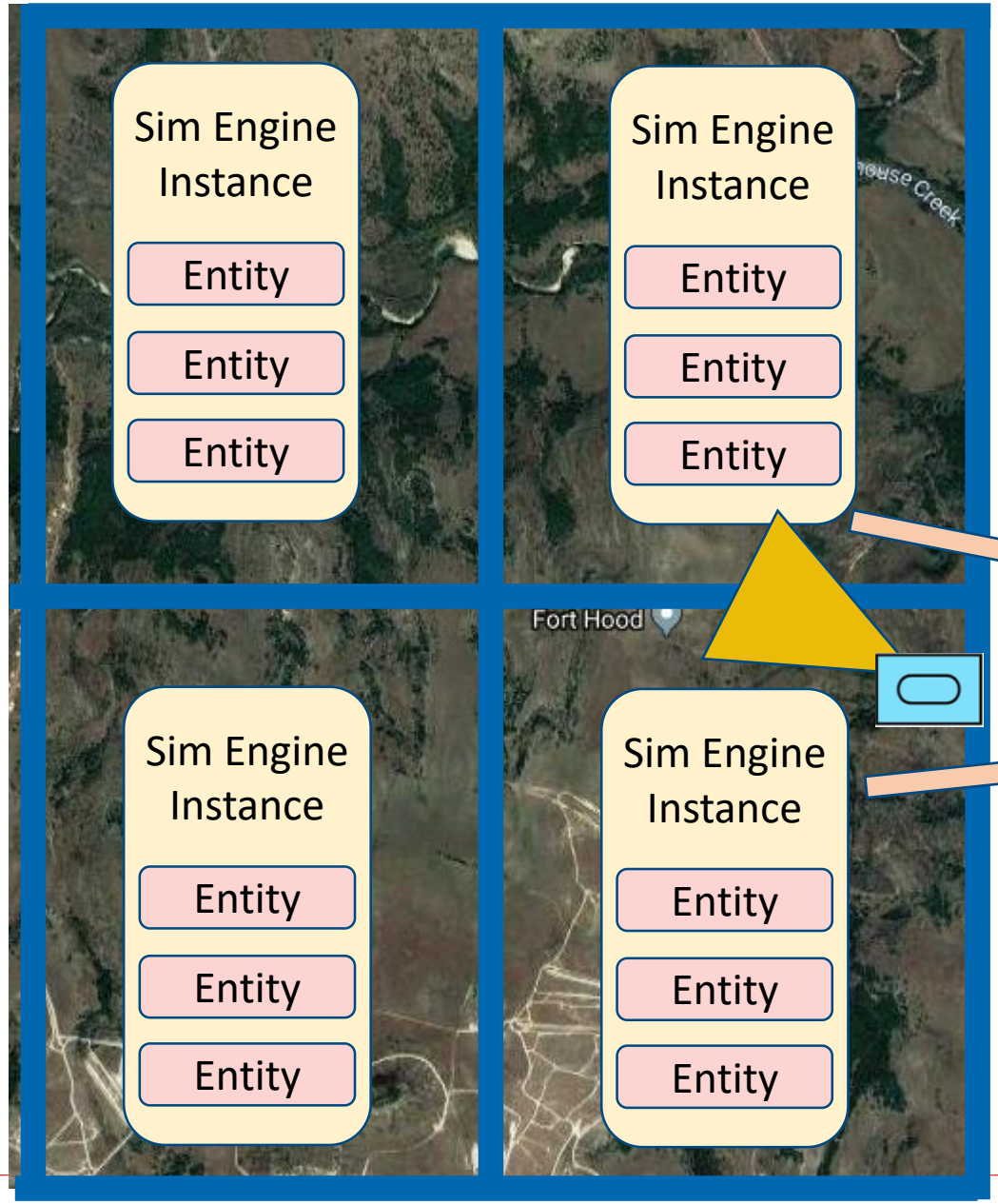
- Divide responsibility
- Communicate state locally or over network

Spatial Organization of Entities



- Each Sim Engine simulates entities in a specific geographic region
- Automatic ownership transfer
- Automatic scaling

Interest Management



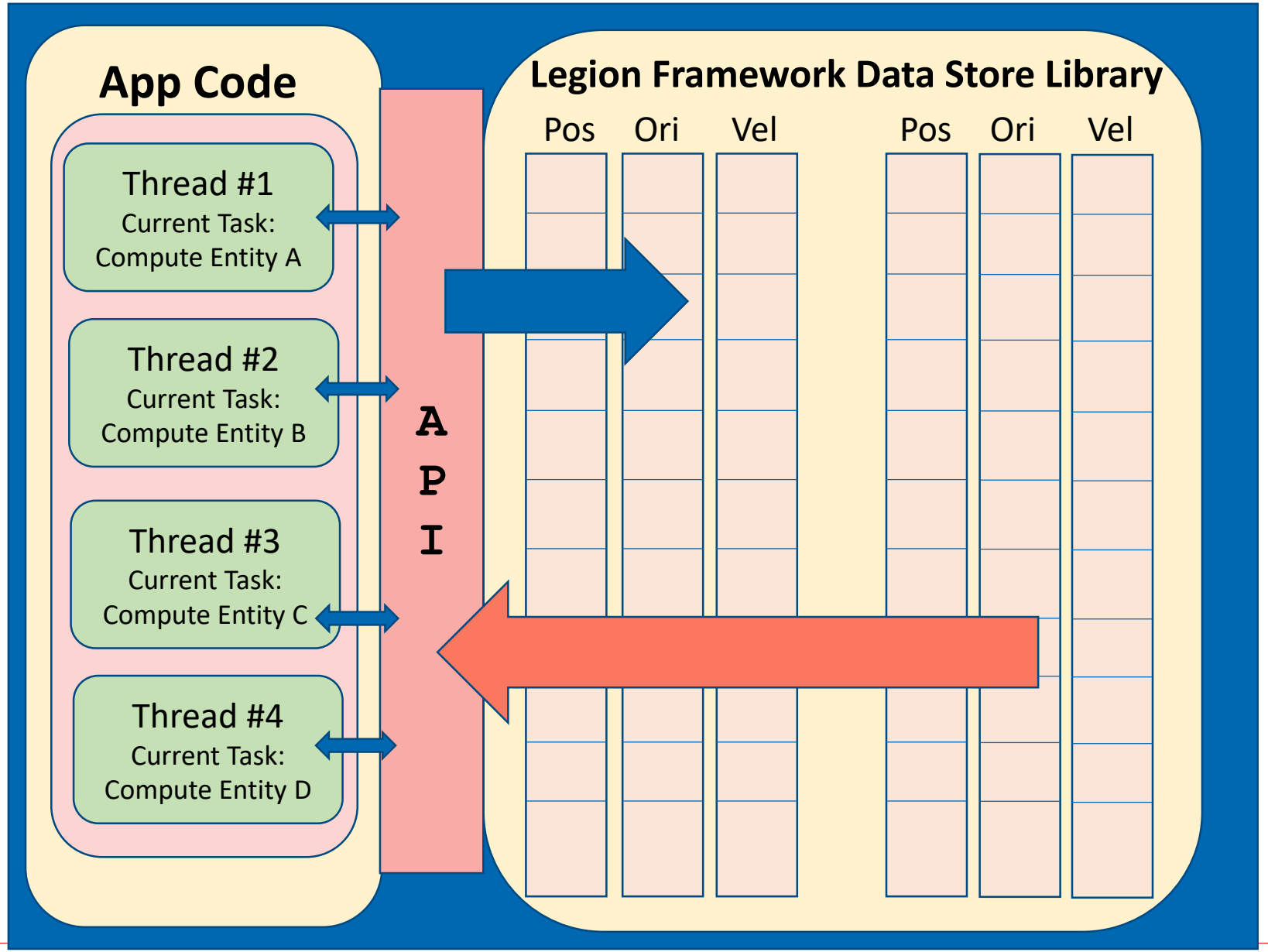
Virtual simulators can only handle small subset of entities in scenario

- Register interest in entities
- Deliver only entities meeting those criteria

Virtual Simulator Interface

- Sim Engine instances also register interest only in entities that they might need to interact with

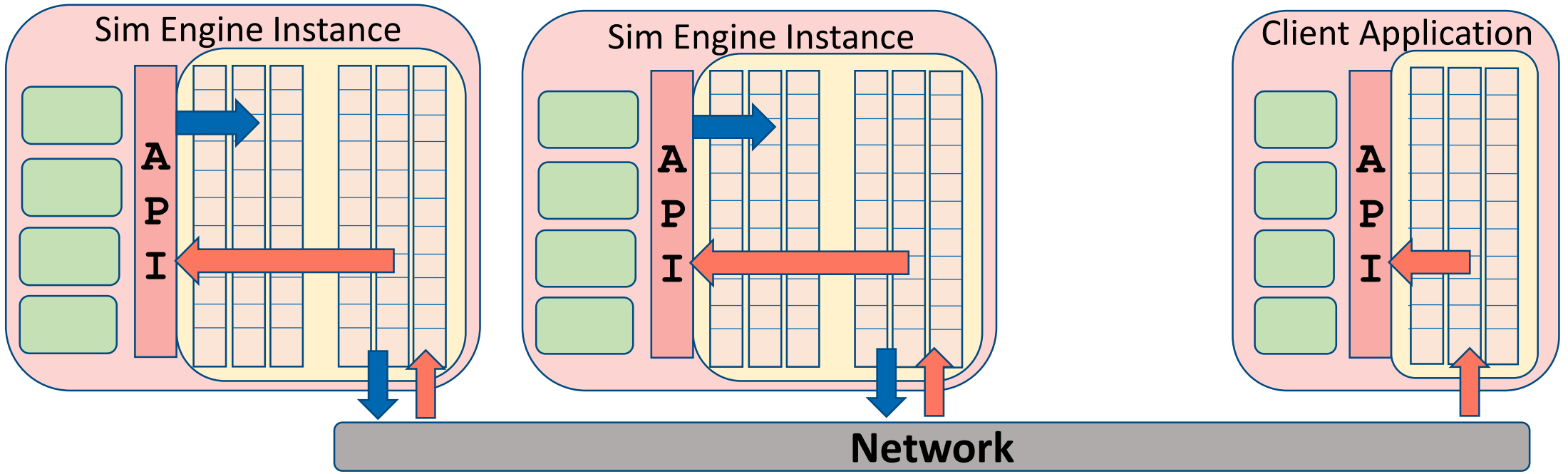
Data-Oriented Implementation / Object-Oriented API



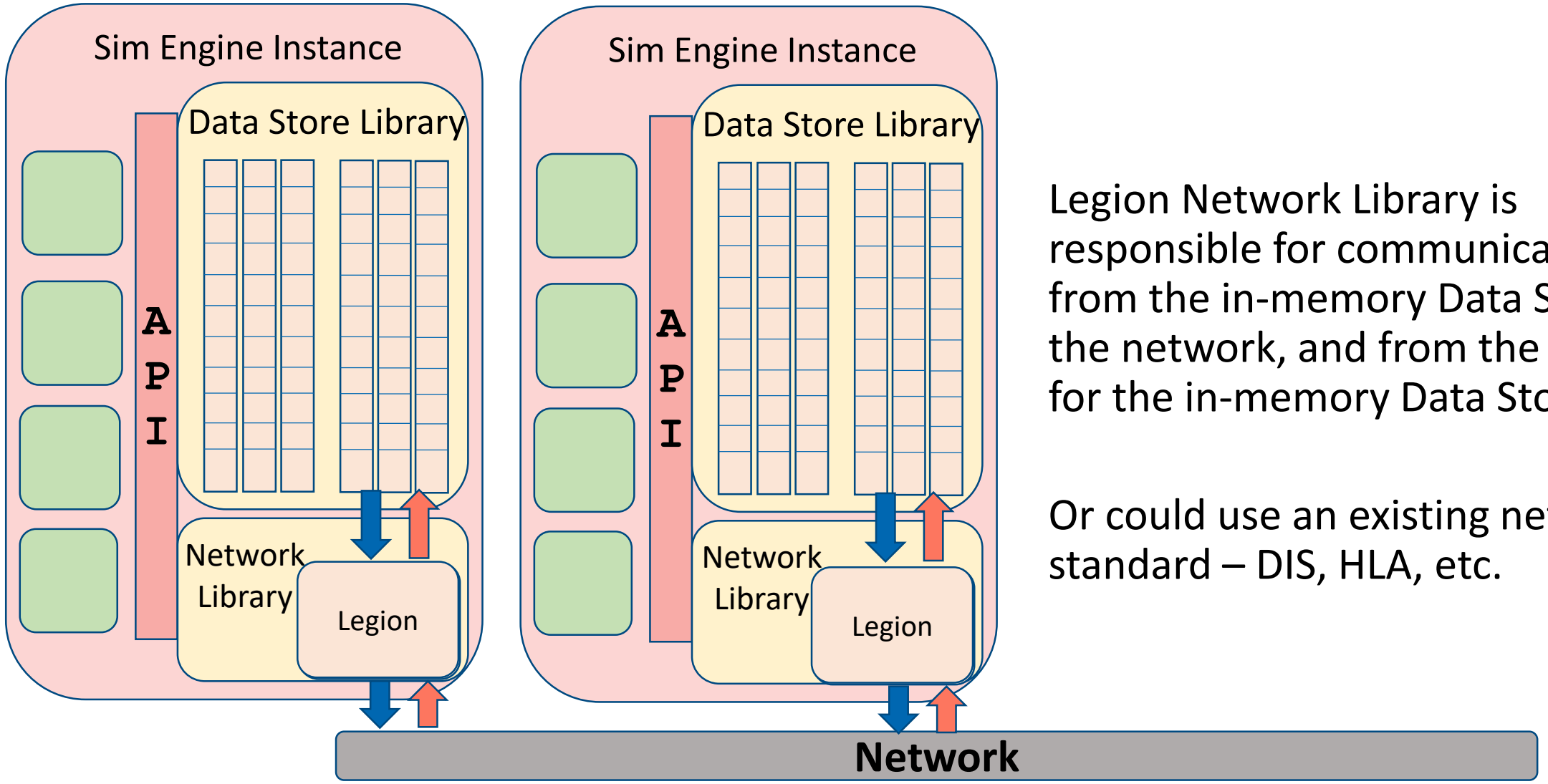
Networking

Now that Sim Engine instances have a well-defined, optimized way of storing the state of objects that it simulates, we need to communicate that data to other applications

We also need a way of populating each application's Data Store with data about entities received from *other* applications



Legion Network Library



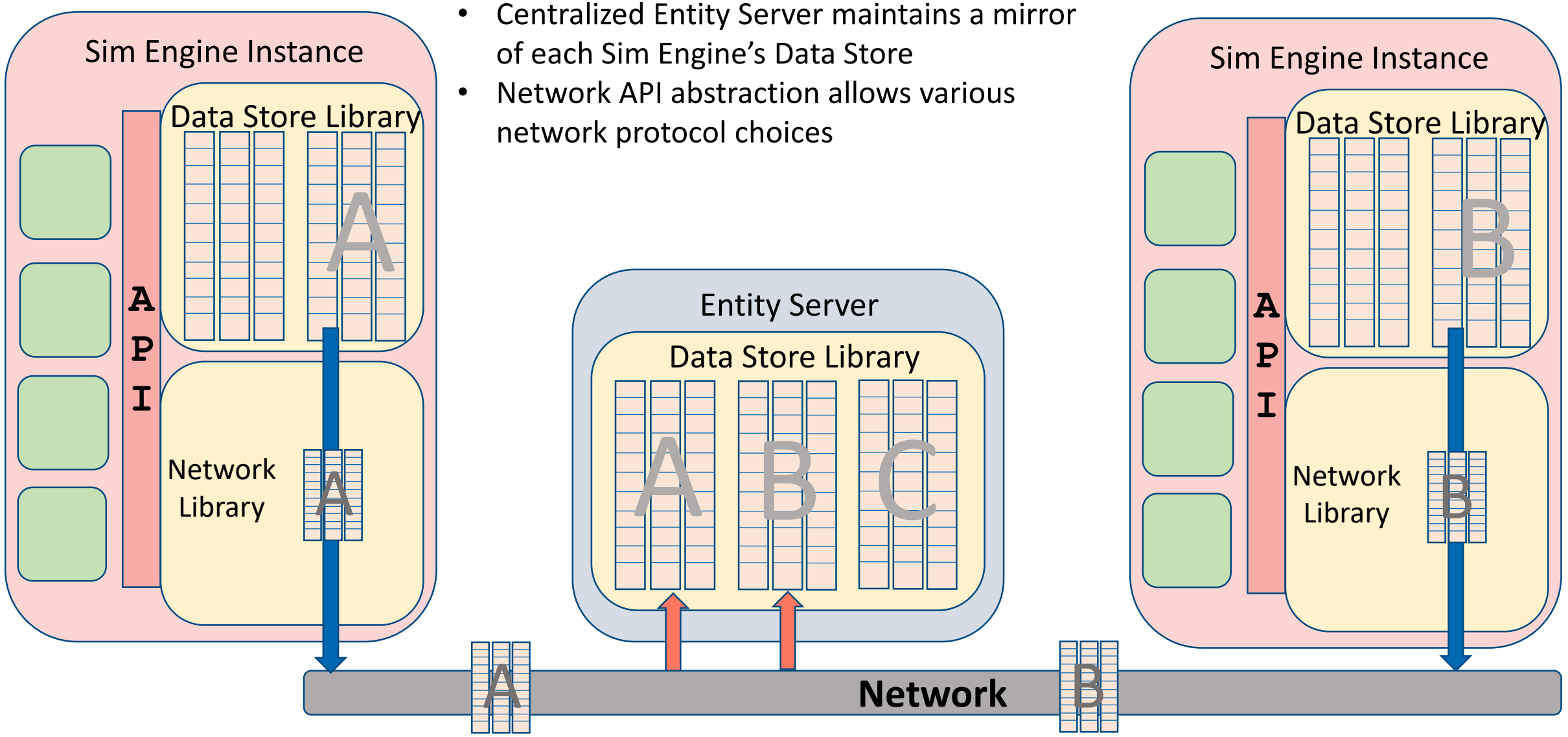
Legion Network Library is responsible for communicating data from the in-memory Data Store to the network, and from the network for the in-memory Data Store

Or could use an existing network standard – DIS, HLA, etc.

Design of MAK's "Legion" Network Architecture

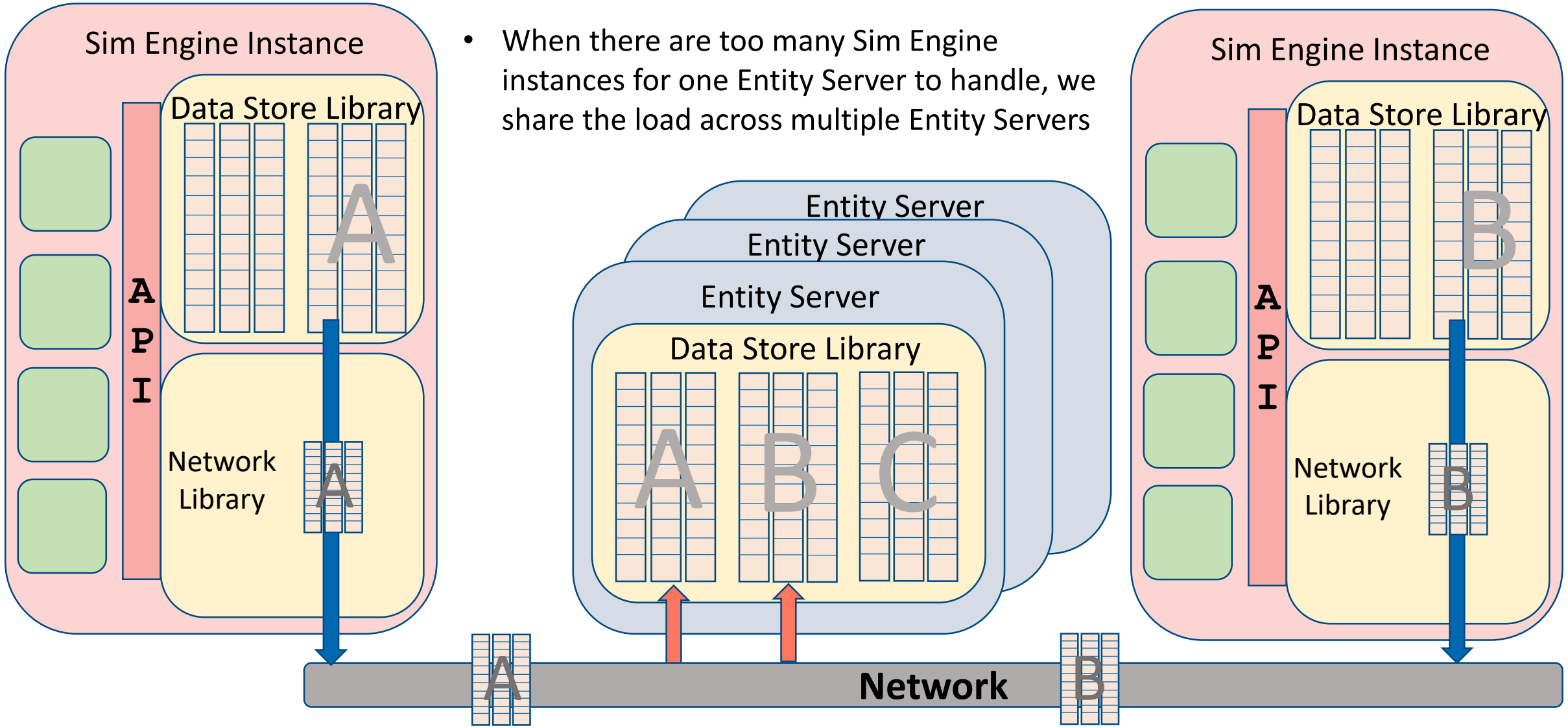
Network System Design

- Centralized Entity Server maintains a mirror of each Sim Engine's Data Store
- Network API abstraction allows various network protocol choices

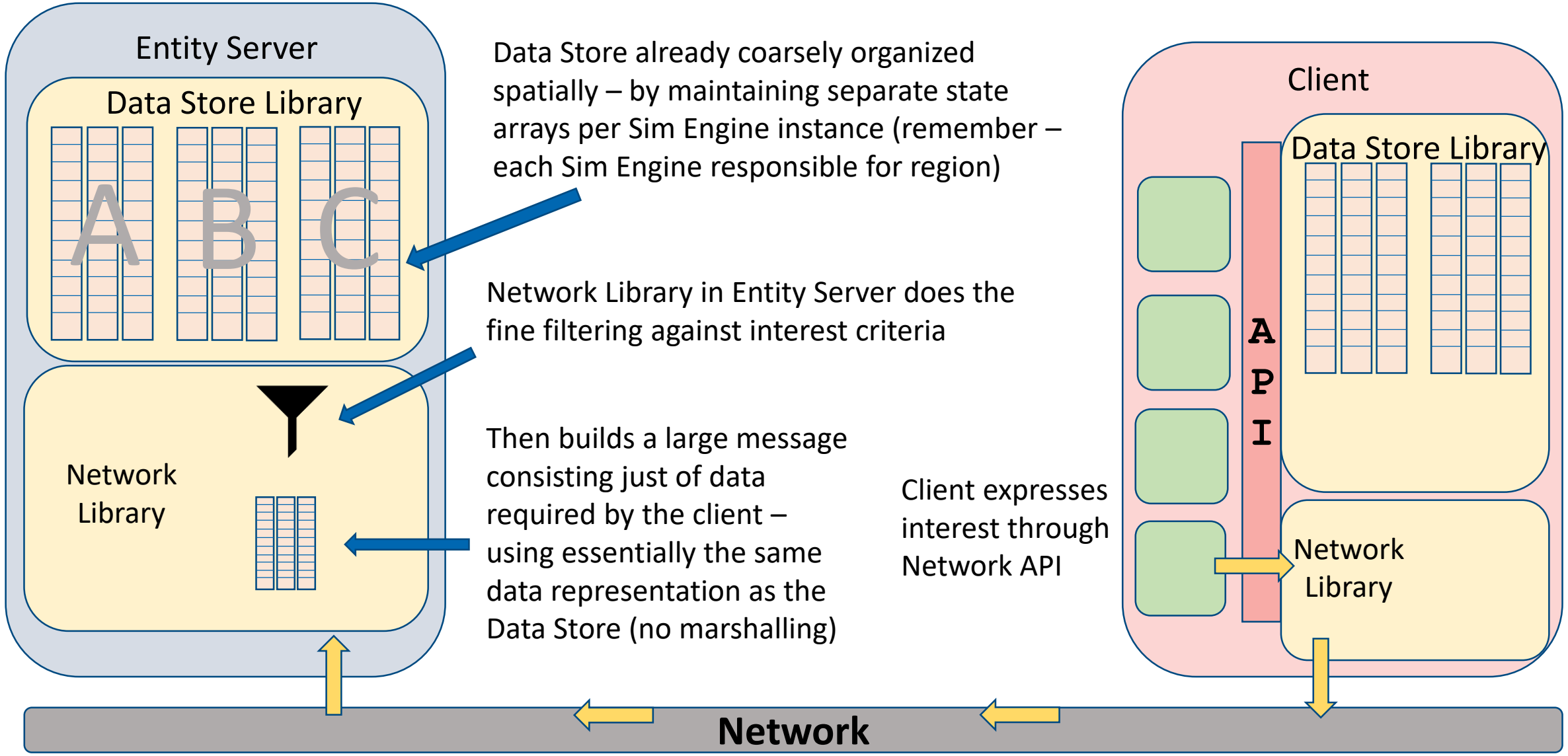


Network System Design

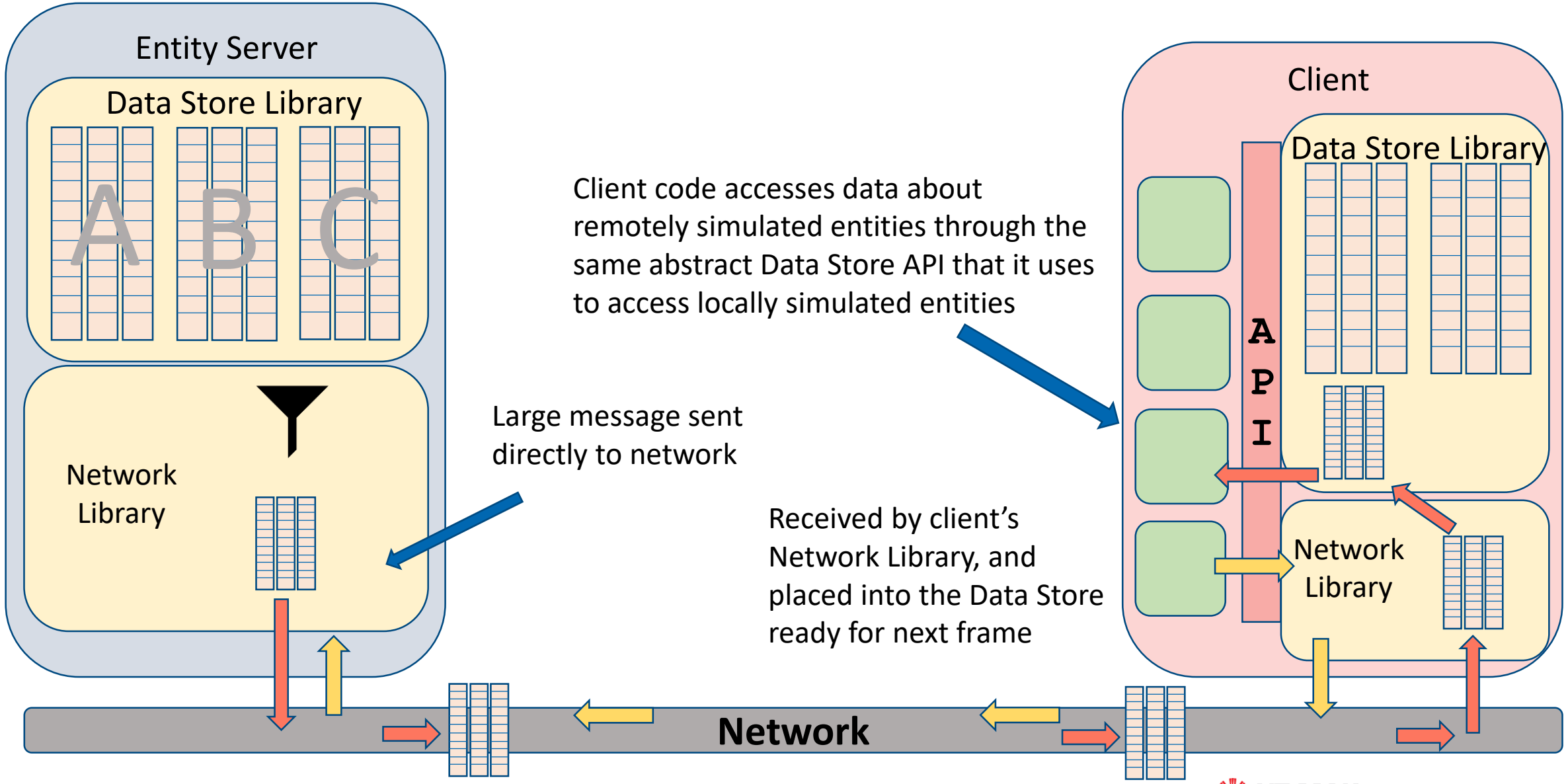
- When there are too many Sim Engine instances for one Entity Server to handle, we share the load across multiple Entity Servers



Network System Design



Network System Design

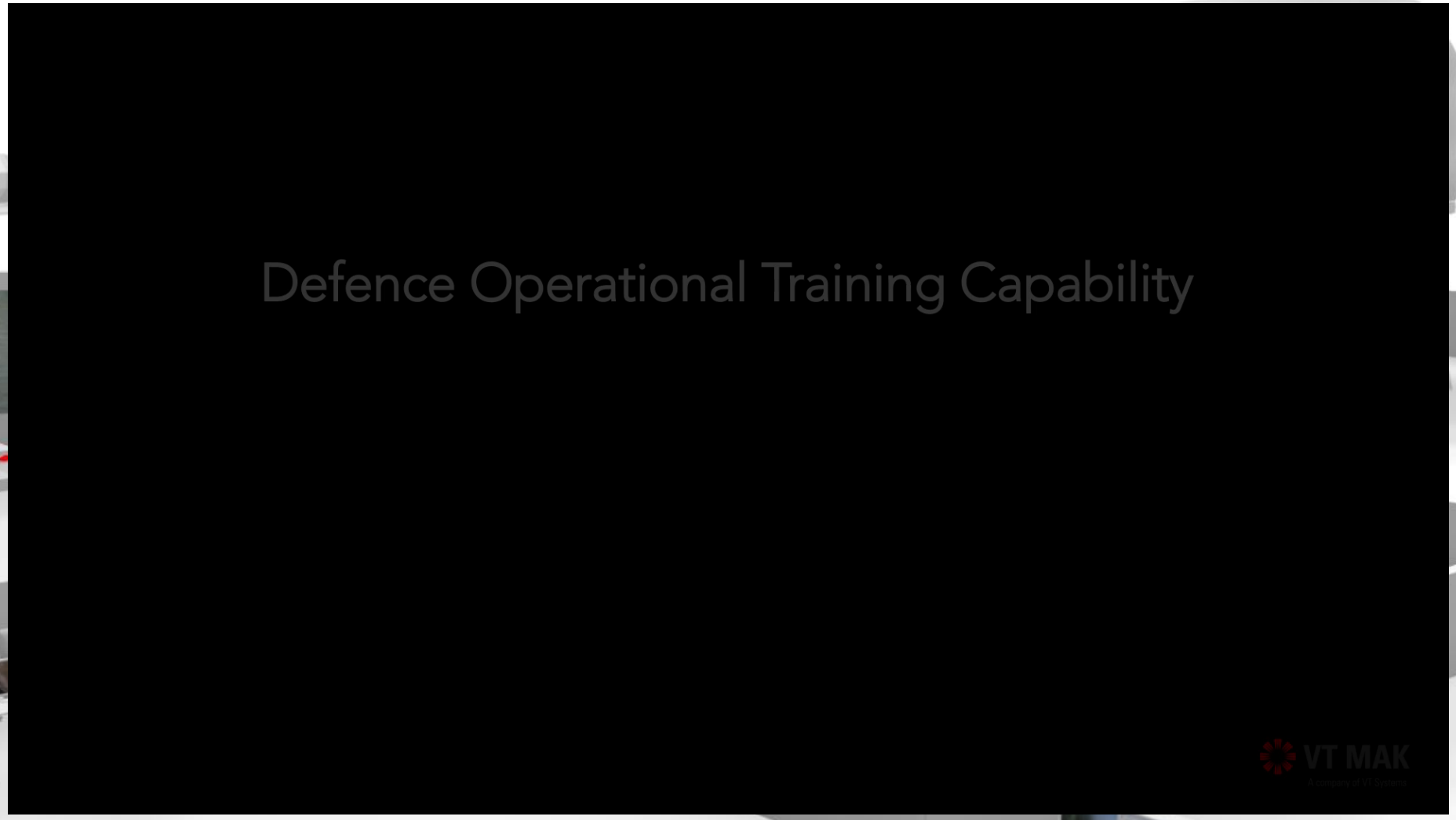


Review of Key Points

- Multiple Sim Engines running in parallel
  - Spatial organization of entities among Sim Engines
  - Ownership transfer to maintain spatial coherence of entities on each engine
  - Interest management to limit how many entities are received by each client
- Separating interface (API) from implementation so that networking specifics can be tailored for use case and network topology
  - Dynamic link compatibility so upgrades can be plugged into applications
- Networking and Data Store designed together to reduce marshalling/copying
  - Same compact binary representations of data used throughout the system
  - Data-oriented design to optimize Sim Engine and support multi-threading
  - Extensibility to allow for custom or extended data models
  - Stateful Entity Server reduces load on Sim Engines
  - Large message with bulk data vastly faster than per-entity packets

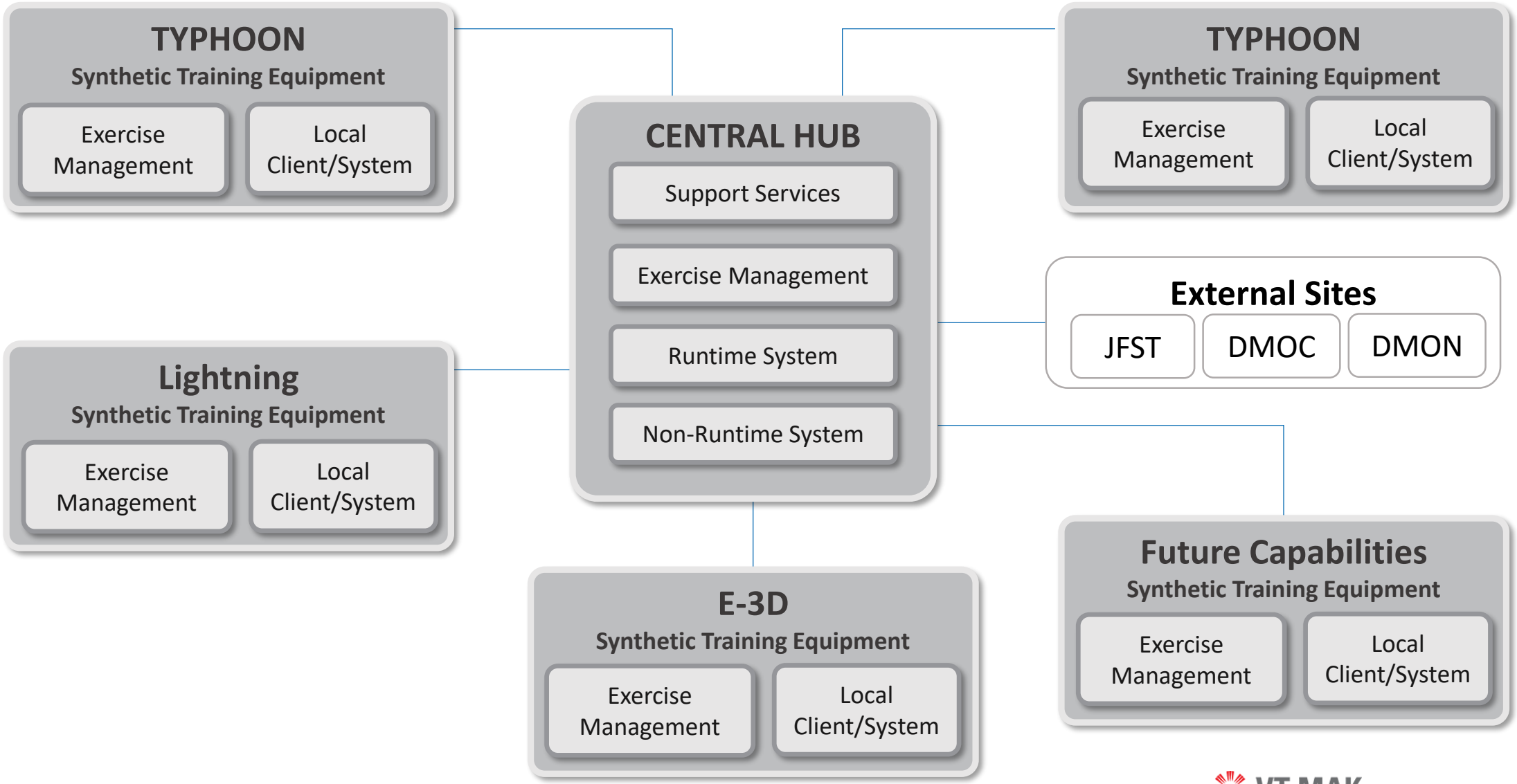
DCS&S

Defence Operational Training Capability (Air) Core System and Services

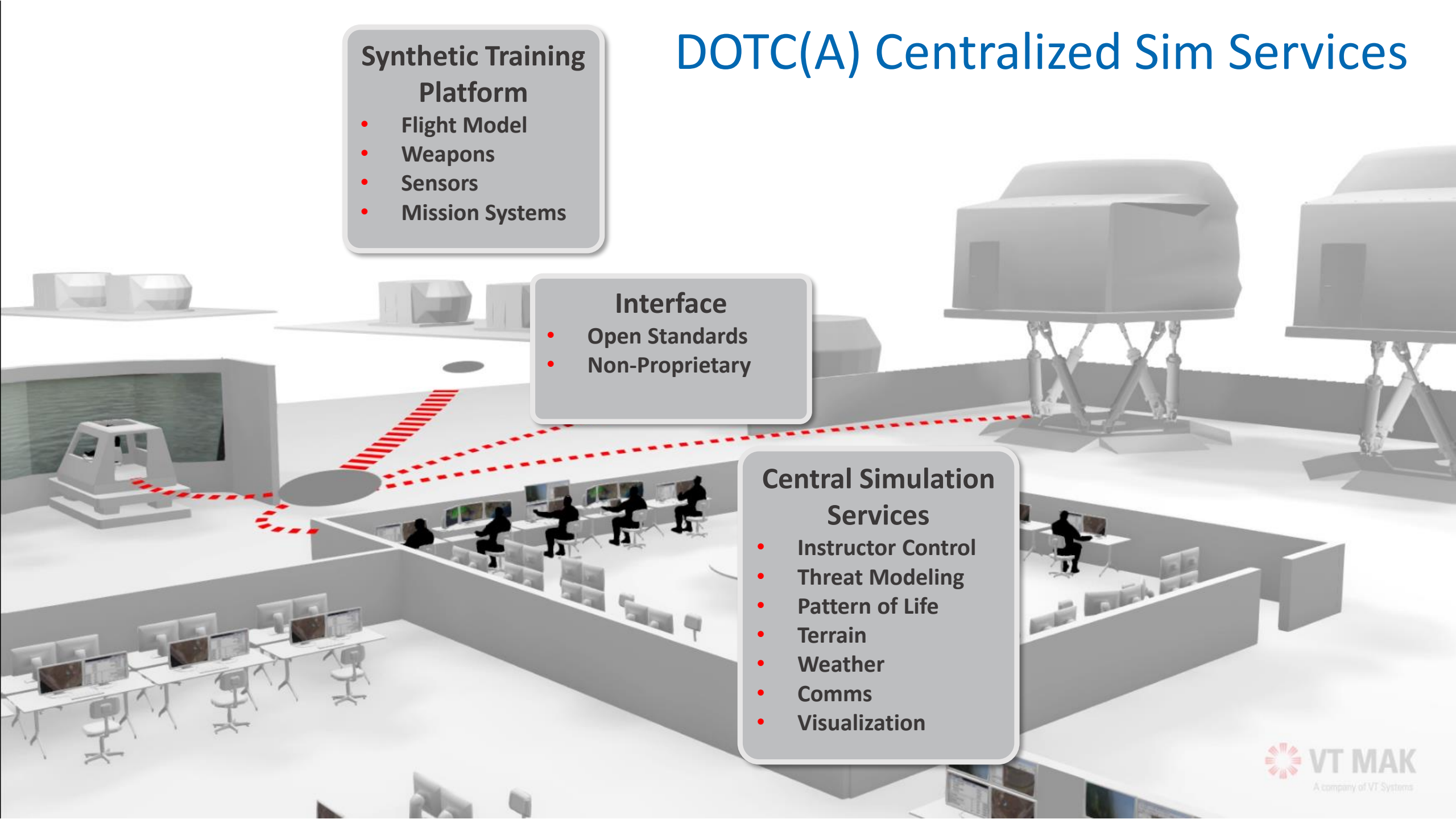


VT MAK
A company of VT Systems

Operational Model



DOTC(A) Centralized Sim Services



Synthetic Training Platform

- Flight Model
- Weapons
- Sensors
- Mission Systems

Interface

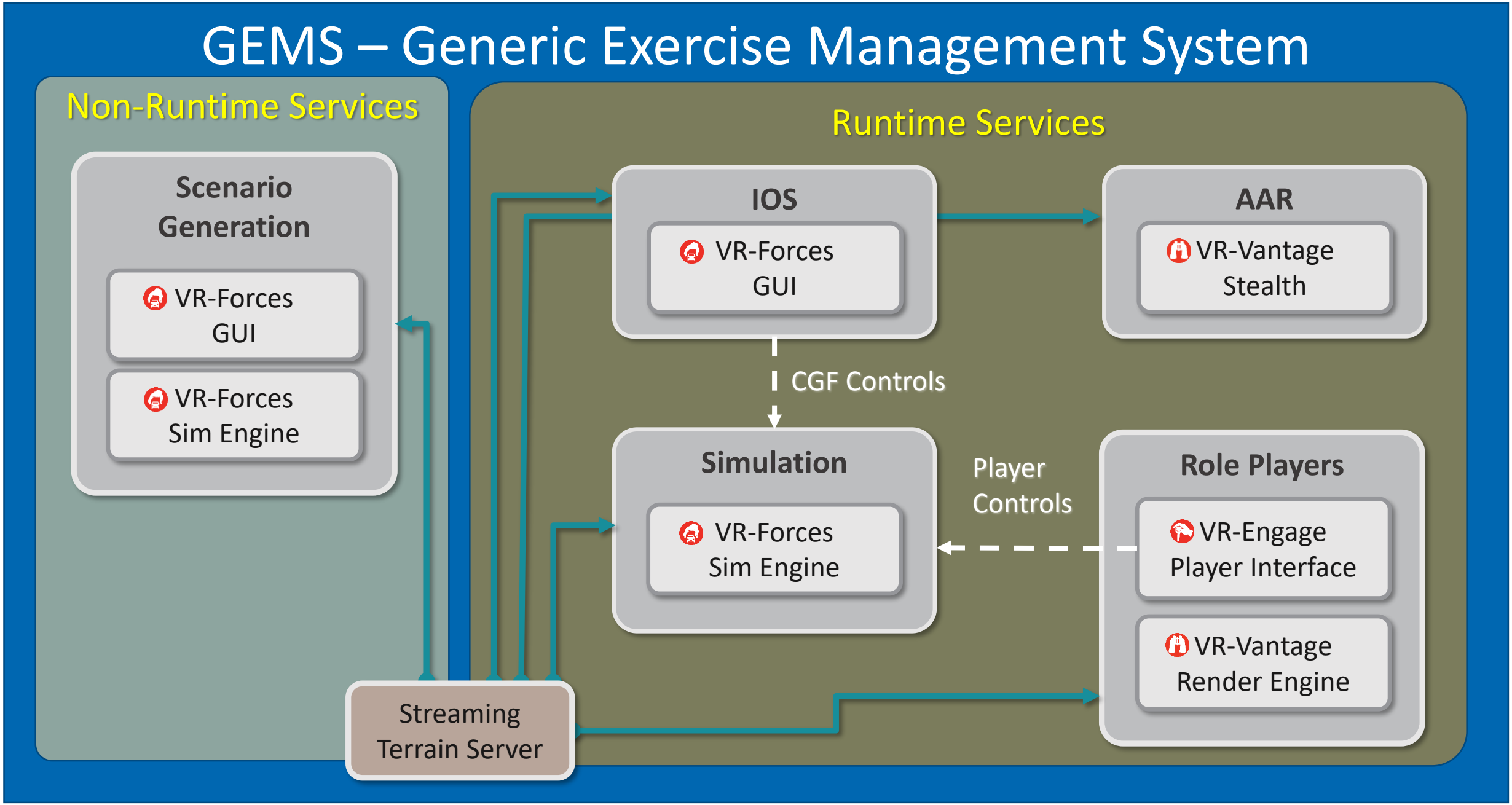
- Open Standards
- Non-Proprietary

Central Simulation Services

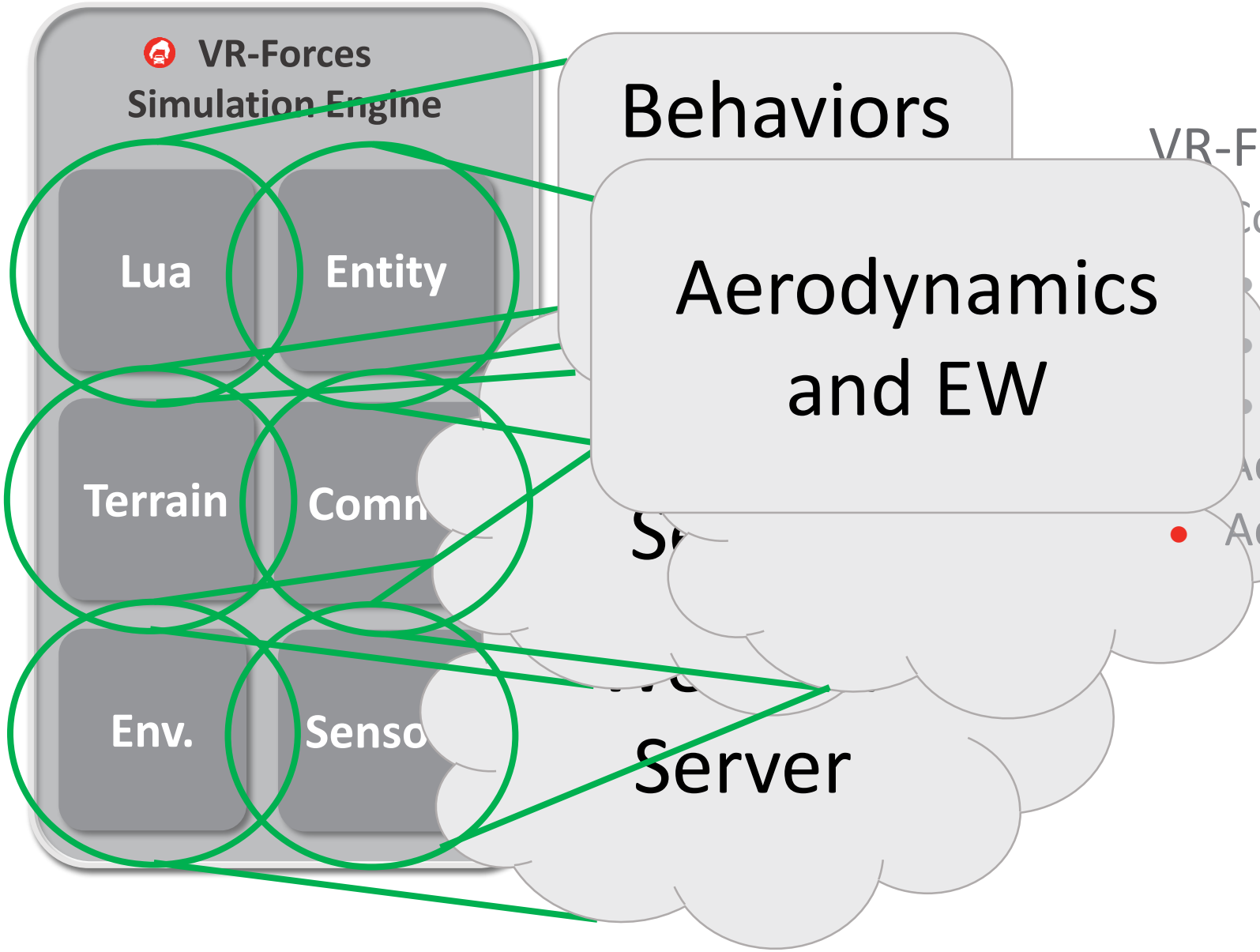
- Instructor Control
- Threat Modeling
- Pattern of Life
- Terrain
- Weather
- Comms
- Visualization

VT MAK
A company of VT Systems

GEMS – Generic Exercise Management System



VR-Forces APIs



VR-Forces APIs will be used to:

- Connect to other services
  - Weather
  - Terrain
  - Remote Control
- Add local content
- Add ITAR capabilities

Conclusion

MAK products are being used in 2 different ways on two major programs:

- As a suite of tools to develop an entirely new architecture and common simulation software for the US Army's Synthetic Training Environment
- As the COTS core components of a large HLA federation for the UK Royal Air Force DOTC(Air) Program

Future programs, such as the Australian Army's LS Core 2.0, can learn from these programs how best to implement their own architectures based on their specific requirements.