

Satellite Toolkit (STK) Integration : AEROPVS031

vsTASKER integrates with **Systems Tool Kit (STK)** by either generating asynchronous commands to control STK scenarios externally or combining C++ generated code with STK's static data. This enables users to perform military and tactical simulations while leveraging precise STK orbit and constellation data.

Key Integration Methods

vsTASKER provides flexible approaches for connecting with STK:

- **Command Generation:** vsTASKER generates commands sent directly to STK with asynchronous results to drive the simulation state.
- **Code Combination:** vsTASKER combines generated C++ code with STK static or dynamic resources.

For tips on how to build satellite constellations and define orbits within STK, which can be integrated into broader vsTASKER scenarios:

Benefits of vsTASKER and STK

By bridging these tools, you can run detailed space and mission scenarios:

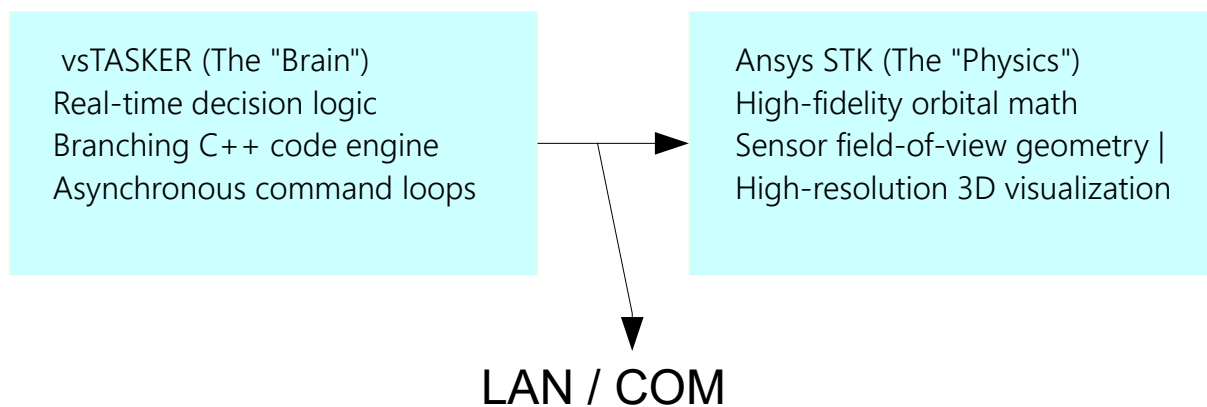
- **Tactical and Mission Logic:** vsTASKER manages the high-level logic, user tactics, and procedural testing.
- **High-Fidelity Space Environments:** STK handles the highly accurate orbital mechanics, space environment modeling, and visibility analyses.
- **Real-Time Visualization:** vsTASKER can easily connect to game engines (e.g., via CIGI) while utilizing STK for precise space data.

Would you like to explore **vsTASKER's native C++ code generation** or discuss setting up **STK/Connect** for remote scenario control?

Integration between Ansys Systems Tool Kit (STK) and VirtualSim vsTASKER simplifies complex mission simulations. **vsTASKER injects real-time dynamic decision-making logic** into STK's high-fidelity spatial and sensor calculations.

Overview of the Integration

While STK natively excels at orbit propagation, sensor simulation, and line-of-sight geometry, it lacks real-time branching logic and conditional decision-making frameworks. VirtualSim vsTASKER acts as the **tactical brain**, allowing users to design visual logic diagrams that dynamically control STK assets.



Supported Integration Schemes

vsTASKER natively integrates with STK using three primary Application Programming Interfaces (APIs) provided by AGI:

- **Local Area Network (LAN):** Connects separate simulation host nodes asynchronously.
- **ActiveX:** Embeds UI components or executes local control.
- **Component Object Model (COM):** Interconnects windows applications directly.

How the Technical Execution Works

The integration relies on a automated hybrid code generation pipeline:

1. **Command Generation:** vsTASKER constructs structured command strings or API calls during runtime.

2. **Library Linking:** The engine combines generated C++ code directly with STK static or exported libraries.
3. **Asynchronous Execution:** Commands pass seamlessly to STK, which returns computational or spatial results back to the vsTASKER loop.

Common Mission Use Cases

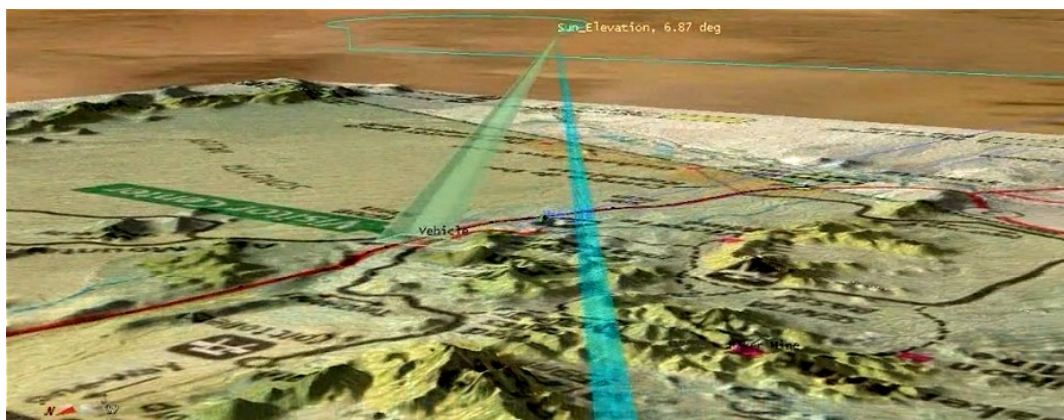
Developers combine both toolkits to handle dynamic scenario changes:

- **Dynamic Interception:** Guiding a missile or aircraft trajectory response based on target tracking data.
- **Responsive Downlinks:** Changing data transmission states or pointing angles when a satellite enters the line-of-sight of a specific ground station.
- **Reactive Payload Controls:** Activating or deactivating optical sensors based on changing weather conditions or target signatures.

If you are setting up this simulation architecture, let me know:

- Which **API scheme** (COM, ActiveX, or LAN) fits your infrastructure best?
- What specific **tactical logic** or branching decisions do your STK assets need to execute?

I can provide sample structural guidance based on your selected setup.



Satellite Toolkit (STK) Integration

by VirtualSim

STK from AGI is a powerful tool for sensor simulation and orbit calculation. However, when it gets to real time simulation with branches, conditions, decisions making (...) vsTASKER make the process simpler and more flexible.

vsTASKER supports natively some integration schemes provided by STK:

- Connect using LAN
- Active/X
- COM

With each API, vsTASKER can either generates commands to be sent to STK with asynchronous results, or combines generated code with STK static or exported libraries.

User can then use logic to activate sensors, change the trajectory of an aircraft or missile, send messages to satellites when in sight from a ground station, etc.

In some projects combining both products, vsTASKER was used as a first sensor detection filter before fine tuning the detection using STK sensors over terrain. Result was returned to vsTASKER for decision making or simple storing in a SQL database for offline analysis.

Similar projects:

-  Using a UAV as a forward observer
-  Sensor assessment on various terrains
-  Event driven STK simulation

OS & HARDWARE SUPPORT	INDUSTRY APPLICATION

Note: ** Specifications are copied from manufacturer data. Please contact before buying.