

VBS-3 integration using ASI Interface : AEROPVS026

Integrating VBS-3 with **vsTASKER** utilizes the Application Scripting Interface (ASI). It operates as a master-slave setup where vsTASKER acts as the logic engine, handling AI, doctrine, and behavior, while VBS-3 handles 3D rendering and platform dynamics.

Key Details & Workflow

- **Communication Link:** vsTASKER communicates with VBS-3 over a local area network (LAN) using a custom plugin (vsTaskerPlugin.dll).
- **Compilation:** This DLL must be compiled and placed in your VBS-3 plugins directory (e.g., VBS3/plugins64) before running the scenario.
- **Setup & Execution:**
 1. You build the scenario and entity logic visually in vsTASKER.
 2. When vsTASKER starts, VBS-3 automatically instantiates the platforms.
 3. vsTASKER uses the ASI interface to inject VBS-3 script commands and updates.
 4. Closing the vsTASKER simulation automatically clears the platforms in VBS-3.

Integration Limitations

- **Performance:** Because the ASI protocol was not designed for heavy, multi-agent real-time data streaming, the ASI interface can be resource-heavy. vsTASKER is best used to populate specific high-level entities rather than syncing thousands of units.
- **Source Access:** The integration logic is fully extendable, and VirtualSim provides vsTASKER with the plugin's source code so users can manually optimize scripts for their use case.

You can read more about the software's capabilities on the official VirtualSim VBS3 ASI Integration page.

If you'd like, let me know:

- Is this for an **air, land, or maritime** scenario?
- What is the **total number of entities** you need to track simultaneously?
- Are you looking for information on vsTASKER's **CIGI (Computer Image Generator Interface)** for advanced graphics?



VBS-3 integration using ASI Interface

by VirtualSim

vsTASKER 6.1 provides a gateway to VBS-3 using a LAN communication with a special plugin loaded by VBS3.

Based on the ASI interface, vsTASKER will send VBS3 script commands and will receive update data for all entities running in the VBS3 simulation for synchronization. It is then possible to let vsTASKER handle some parts of a scenario while leaving VBS3 handling the platform dynamics, the line of sight, or the exchange fire and damages. vsTASKER will enable and disable the AI according to situations and time. The end user will gain flexibility and control while building a specific scenario.

Using vsTASKER, all the scripts commands and functions of VBS3 can be encapsulated into actions, tasks, logic of vsTASKER and attached to any scenario entity on a simple mouse click. When running the simulation, logic can be visually monitored in real-time.

Because vsTASKER simulation engine and the ASI plugin are disconnected, the scenario development is quick: every time vsTASKER SIM starts, VBS3 creates locally

all platforms and units defined in the scenario, then wait for commands to be unfold through the ASI interface. When vsTASKER stops, all VBS3 platforms and units are removed.

Features:

- ASI interfacing for better independence
- Drag and drop of predefined logic
- Easy set up of a complex scenario without using scripts
- Quick tries and errors with no VBS3 reload time
- No runtime license for any deployed scenario built with vsTASKER
- Avoiding the VBS3 script debugger makes building complex scenarios simpler
- Easy mix of VBS3 AI and custom designed vsTASKER doctrines
- Several modes to drive VBS3 entities from vsTASKER

OS & HARDWARE SUPPORT	INDUSTRY APPLICATION

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