

Radar simulation for Maritime Surveillance : AEROPVS019

vsTASKER is a real-time simulation toolkit developed by **VirtualSim** that allows engineers to design, build, and deploy complex logic and scenarios, including advanced **radar simulation for maritime surveillance**. In maritime surveillance applications, it is typically used as the primary engine to script, generate, and run the physical movements and logic of the entire tactical environment.

How vsTASKER is Used in Maritime Surveillance

The software manages the entire workflow of a maritime surveillance exercise, transforming geospatial data into live simulation logic:

- **Scenario Generation:** Users define the underlying environment using official **S-57 electronic navigational charts**. Coastal structures, shallow water boundaries, and terrain features are loaded directly to establish realistic line-of-sight constraints.
- **Asset and Threat Injection:** Operators place assets (e.g., coastal radar stations, maritime patrol aircraft, drones) and targets (e.g., cargo ships, smuggling boats, oil pollution threats) into the workspace.
- **Real-Time Data Generation:** As the simulation runs, vsTASKER tracks the trajectories of all moving objects, applies simple or advanced radar equations, and computes radar detections dynamically.

Integration with vsRAD for Visualisation

In an actual development or training deployment, vsTASKER functions as the backend simulation engine, which then pairs with a dedicated radar display toolkit:

1. **The Backend (vsTASKER):** Simulates the radar physics, computes target positions, handles blind sectors or clutter, and outputs raw detection data.
2. **The Frontend (vsRAD):** A C-written, OpenGL-based graphical library that takes vsTASKER's real-time detection data and maps it onto a realistic **Plan Position Indicator (PPI) screen** (a traditional sweeping radar view).

3. **Customisation:** Developers use the open API of the vsRAD library to integrate vsTASKER's radar feeds directly into custom operator panels, command-and-control (C2) mock-ups, or wider data-fusion systems.

Core Technical Advantages

- **Automatic Code Generation:** It features a graphical interface for drawing behavior logic, which then automatically compiles into standalone C++ simulation engines.
- **High Scalability:** It supports batch scenario runs—often scaling into millions of variations—enabling developers to generate massive datasets for testing automated target tracking algorithms or artificial intelligence systems.
- **Interoperability:** The platform easily communicates over standard simulation protocols like High Level Architecture (HLA) to sync with other military or maritime simulators.

If you are currently setting up a scenario, I can help you outline the **mathematical radar equations** to use, configure **S-57 chart data logic**, or structure the **C++ code deployment**. Which of these would you like to explore next?

Using **vsTASKER**, a leading C2 and scenario-generation software, you can execute advanced maritime surveillance radar simulations. By combining it with 3D visualization or visualizer toolkits like **vsRAD**, vsTASKER acts as the scenario engine to generate real-time radar detections based on S-57 navigation charts and DTED terrain data.

Setting Up Your Maritime Scenario

To configure your environment effectively, you can set up the simulation with specific parameters:

- **Scenario Definition:** Map out your operating area using native S-57 electronic navigation charts and DTED (Digital Terrain Elevation Data) for accurate shoreline and coastal topology modeling.
- **Assets & Threats:** Program your assets (e.g., airborne platforms, coastal stations, patrol vessels) and threats (e.g., smuggling boats, rogue vessels, or oil spills) along with their respective speeds, trajectories, and detectability.
- **Radar Placement:** Place fixed coastal radars (such as an X-band or Ku-band system) on precise geographical coordinates.

The vsTASKER & vsRAD Pipeline

The simulation generates raw data through an integrated workflow:

1. **Signal Modeling:** vsTASKER utilizes spherical Earth models and standard radar equations to calculate the dynamic behavior of targets, taking into account horizon limits and topographical blockages.
2. **Data Processing:** vsTASKER then feeds the resulting radar detection arrays in real-time into the **vsRAD** toolkit (or a similar OpenGL-based visualizer) to be rendered as realistic sweeping images.
3. **Tracking & Fusion:** Modern implementations can pass these detections into advanced filter algorithms (e.g., Track-Before-Detect) or overlay them with other feeds like AIS (Automatic Identification System) for multi-sensor fusion.

Key Capabilities and Features

- **Environmental Factors:** You can test hardware limitations in varying conditions by simulating rain or rough sea states, enabling you to practice applying rain and sea clutter controls.
- **Real-time API:** The vsRAD engine provides a complete and open API, allowing you to easily adapt and customize the display panels for integration into a wider user command panel.

Would you like to explore:

- Integrating **AIS and Radar data fusion**?
- Specific **sea clutter and RCS (Radar Cross-Section) modeling** for ships?
- Connecting this setup to a **GIS or Google Earth** interface?

Radar simulation for Maritime Surveillance

by VirtualSim

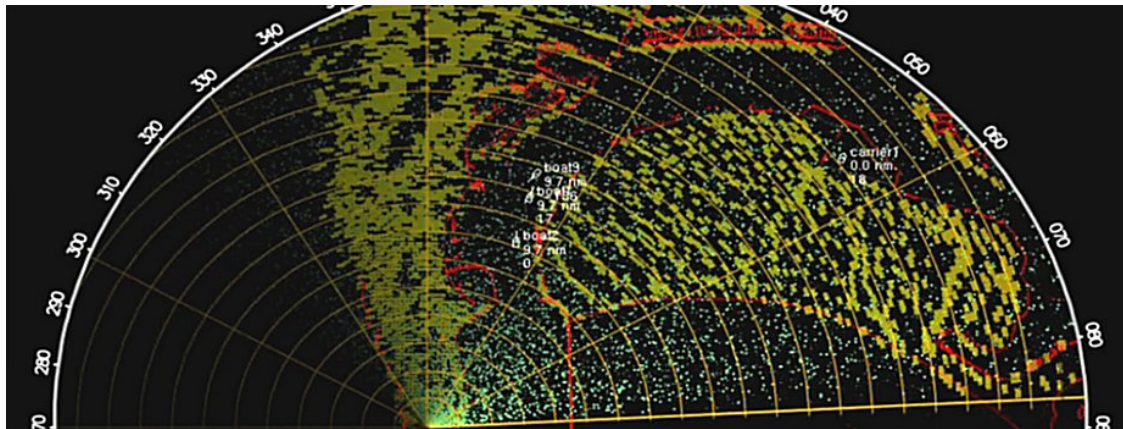
The project was requesting a radar visualizer toolkit to be fully adaptable to be integrated into a much bigger user panel. vsRAD is written in C and based on OpenGL. It is provided as a library with a complete and open API to customize every aspect of the display.

In this solution, vsTASKER was used to setup the maritime scenario, based on S57 charts. The coastal radar was put on the Old Point Loma lighthouse. vsTASKER is generating the radar detections send them to vsRAD for display in real-time.

The default vsRAD graphical interface was used for the mock-up only. The final panel integrated all the functionalities in core.

What was needed:

- 📁 vsRAD radar visualizer toolkit
- 📁 San Diego S57 navigation chart
- 📁 California DTED 2 for the terrain elevations
- 📁 vsTASKER for the scenario generation and radar detection
- 📁 A specific component for binding vsRAD and vsTASKER



OS & HARDWARE SUPPORT	INDUSTRY APPLICATION

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