

NASA JPL

The Design and Architecture of the Rover Sequencing and Visualization Program (RSVP)

SpaceOps 2004 Presentation

Scott Maxwell
 Brian Cooper
 Frank Hartman
 John Wright
 Jeng Yen

NASA JPL

Software Design Overview

```

    graph LR
      HyperDrive[HyperDrive] --- Executive((Executive))
      RoSE[RoSE] --- Executive
      Executive --- SequenceFlow[Sequence Flow]
      Executive --- ImageBrowser[Image Browser]
      Executive --- Logger[Logger]
  
```

- RSVP is the MER project's command-level editor
 - Textual command editing
 - Graphical command editing
 - Simulation, resource modeling, error checking, etc.
- Central system design feature is a message-passing hub (the Executive)
- RSVP component applications run independently, plug into the Executive, and coordinate work by passing messages
- Message-passing library based on PVM (thanks, Oak Ridge National Laboratory!)

SAM-2 12/28/03

NASA JPL

RoSE component

- Rover Sequence Editor
- Textual command editor
- Aggressively data-driven
 - Spacecraft command dictionary is just another config file
 - Dynamically constructs command-editing GUIs from command dictionary read at startup time
- "First among equals"
 - Command ID assignment
 - Provides interface to launch other RSVP apps
- Save/load sequence in XML-based RML format (native format) and legacy formats for use with other JPL tools
- Display resource modeling info

SAM-3 12/28/03

NASA JPL

HyperDrive component

- 3-D command editor
- Displays terrain
- Move rover within terrain
- Specialized for editing a subset of MER commands:
 - Driving
 - Arm motion
 - Imaging
- Simulate rover's behavior
- Store and play back simulation results as a "movie"
 - Valuable review product
 - Valuable outreach product

SAM-4 12/28/03

NASA **Other RSVP components** JPL

- **Logger**
 - Simple ASCII dump of messages flowing through the system
- **Image Browser**
 - Show mono/stereo images downlinked from rover
 - Stereo images shown in stereo modes (3-D goggles or "cross-eyed")
- **Sequence Flow Browser**
 - Graphically displays timeline for each sequence
 - Commands appear on the timeline where they execute
 - Overlays comm windows when info available
- **Other applications as needed**
 - Modular design: add new functionality by writing a new component app; plug it in by adding it to a configuration file

SAA-4 12/28/03

NASA **Message-passing example #1 (adding a command)** JPL

1. Component app sends ADD_CMD to Executive with special -1 command ID
2. Executive notices -1 ID, sends to RoSE
3. RoSE assigns ID (here, 42), adds command internally, passes resulting message back to Executive
4. Executive broadcasts message to all apps

When RoSE itself adds a command, process starts at step #3.

SAA-6 12/28/03

NASA **Message-passing example #2 (deleting a command)** JPL

1. Component app sends DEL_CMD message with ID of command to delete
 - XML text for command is also included for the benefit of human readers; not needed by tools
2. Executive broadcasts DEL_CMD message to all other tools

Most cases are as simple as this one; the ADD_CMD case is unusual

SAA-7 12/28/03

NASA **All message types** JPL

Message	Description	Message	Description
ADD_CMD	Add a command	MOD_REGION	Modify a range region
DEL_CMD	Delete command	SET_REFERENCE_POINT	Set a reference point
MOD_CMD	Modify command	SET_XPM	We don't remember what this was supposed to be for!
CMD_CONTROL	Random outflows (see RML, MR see component app, etc.)	SEQN_BURSTING	Report Drive-modified command duration to use in SEQNEN
NEW_RML	New RML file	SEQN_TIMES	Command start times resolved from SEQNEN
ADD_RML	Add RML text to file	EXPLORE_FUTURE	Explores a "future" file which a set of conditional commands is assumed to work out in a specified way!
3D_CURSOR	Place 3-D cursor	SELECT_FUTURE	Select a previously explored future
SET_INSERT_PT	Set insertion point for next added command	SET_STATE	Set state data to some value point
LAUNCH_SUBTOOL	Launch nested RSVP component app	IMAGE_DATA	New image data from Image Browser
KILL_SUBTOOLS	Kill all non-RoSE component apps	SEL_CMD	Set of commands to select in tests
ADD_REGION	Add a range region	SEL_ADD	Add to set of selected commands
DEL_REGION	Delete a range region	SC_ID	Spencer's ID and related info

SAA-8 12/28/03



Conclusion: message passing is good



- **Facilitates multi-language development**
 - RoSE, Executive, Logger in Java
 - HyperDrive, Image Viewer, Sequence Flow Browser in C++
 - Other apps could be written in any language that can connect to PVM (Perl, Python, etc.)
- **Can develop components independently**
 - All inter-application interfaces "forced" to be clean
- **Easily add/remove functionality**
- **Send messages across network instead of machine-local to get load-balancing/collaboration "for free"**