



connect  
**OpenVMS**  
Boot Camp 2011

September 18 – 22  
Sheraton Hotel – Needham, MA

# Using OpenVMS Technologies to Build an Agile Computing Base

**Robert Gezelter, CSA, CSE**

**September 2011**

As a courtesy to your fellow attendees, kindly set all devices (electronic and otherwise) to the silent or vibrate mode. If you receive a call, please leave the room to answer it.

The real world:

Target stores, September 2011:

- New limited edition Missoni line introduced
- Consumer surge overwhelmed provisioning
- Site reportedly effectively inaccessible for business day
- Not attack, explosion of popularity

*“As you know, you go to war with the Army you have. They're not the Army you might want or wish to have at a later time.”*

- Defense Secretary Donald Rumsfeld, 2004

## The meaning of agility:

- Fast, sure response to external stimuli
- No time to plan, edit, test
- No time to build new systems to handle surges

This session was inspired by actual client episode:

- Need to add new processing assets to OpenVMS cluster
- Should have been straightforward
- Complications: Instantiation dependencies

## Instantiation Dependencies: An Example

```
$ IF F$GETSYI("NODENAME") .EQS. "FRED"  
$   THEN  
    . . .  
$   ENDF
```

Repeated tens of times, for many different nodes throughout various startup procedures.

## The consequences:

- Adding a node, or rebalancing workload is a time consuming process
- Each node-dependency had to be examined individually
- A multiple man-day, error-prone process



A better way:

Traditional OpenVMS use of SYS\$SPECIFIC:

- Helpful:
  - move specification of dependencies to a single file (e.g., SYS\$SPECIFIC:[SYSMGR])
  - Use the STARTUP database, STARTUP\$STARTUP\_LAYERED.DAT to keep track of options per node

But there is still a shortcoming: Changes within an OpenVMS cluster member:

- Each member traditionally has a single SYS\$SPECIFIC directory
- Files common to multiple members are in SYS\$COMMON
- If there are multiple SYS\$SPECIFIC directories, they are not connected
- Potential for accident and unintended drift

## A Better Solution: Member specific files, to wit: SYS\$NODE\_SPECIFIC

- SYS\$SYSROOT is normally defined during Phase INITIAL by  
`SYS$COMMON: [ SYS$STARTUP ] VMS$INITIAL-050_VMS.COM`
- Create new directory on system disk:  
`SYS$SYSDEVICE: [ NODES ]`
- Create a subdirectory for each OpenVMS cluster node under NODES

Also, create `SYS$SITE_SPECIFIC`

- As `SYS$NODE_SPECIFIC` can be used as a pointer for Node specifics
- Files specific to a particular site in a multi-site OpenVMS cluster

## Ways to define SYS\$NODE\_SPECIFIC and SYS\$SITE\_SPECIFIC

- **Modify**

`SYS$COMMON:[SYS$STARTUP]VMS$INITIAL-050_VMS.COM`

- **Add user-defined file to list in**

`STARTUP$STARTUP_LAYERED.DAT` during phase  
**DEVICES**

## Demonstration: RLGSCDEFINE.COM

- Added to `STARTUP$STARTUP_LAYERED`
- Executed during phase `DEVICES`
- Creates `SYS$NODE_SPECIFIC` logical name
- Inserts node specific directory in `SYS$SYSROOT`

# Demonstration session

## MODPARAMS.DAT

- Use `AGEN$INCLUDE` to include node-specific information:
  - `SCSID`
  - `SCSNODE`
  - Standard configuration parameters
  - Test in `SYS$SPECIFIC`, promote to `SYS$NODE_SPECIFIC`



## MODPARAMS.DAT (continued)

```
AGEN$INCLUDE SYS$NODE_SPECIFIC:[SYSEXE]MODPARAMS.DAT
```

- Is the first line in the SYS\$SPECIFIC file
- That allows “default” node parameters to be overridden as needed
- In effect, one can start with a file only containing the AGEN\$INCLUDE line in the SYS\$SPECIFIC directory

## Creating new nodes

- All variations conditionalized upon logical names or the Startup database
- No IF statements referencing node names
- Special command files in SYS\$NODE\_SPECIFIC
- Contingency command files referred to in SYS\$SPECIFIC

## Emergency procedure: Reboot from pre-defined contingency root

- No emergency editing needed
- Pre-defined, vetted set of options

## Adding members to the cluster

- Copy SYS\$SPECIFIC tree
- Add license units
- Create Node Specific directory
- Define Node specific MODPARAMS.DAT
- Run Autogen

## Adding members (continued)

- Initial boot using spare machine
- “Ready-to-go” members when load increases
- Adding node consists of activating VM, inserting blade, configuring hardware

## Questions

### Supplemental Materials, Slides:

<http://www.rlgsc.com/openvms-bootcamp/2011/agile-openvms.html>