



A World of Difference in
Database Support

Managing OpenVMS As If Your Rdb Database Depended On It!

Brad McCusker
Software Concepts International, LLC
402 Amherst Street, Suite 300
Nashua, NH 03063, USA

A stylized graphic of a globe showing the continents, with a blue arc representing a latitude or longitude line passing through it.

Phone: 603-879-9022
e-mail: mccusker@sciinc.com
www.sciinc.com

Agenda

- ◆ Introduction
- ◆ Current state of OpenVMS Systems
- ◆ A System Management Framework for OpenVMS

Agenda

- ◆ Introduction
- ◆ Current state of OpenVMS Systems
- ◆ A System Management Framework for OpenVMS

About Software Concepts International

- ◆ Located in Nashua, NH (USA)
- ◆ 24+ years in business supporting OpenVMS
- ◆ An international reputation
 - ◆ A leading provider of remote managed DBA services for the Rdb and DBMS databases
 - ◆ A leading provider of remote managed services for OpenVMS systems
- ◆ Proven track record
 - ◆ Actively managing 100s of databases and dozens of systems and configurations
 - ◆ Remote DBA service since 1995 (still supporting many of the same sites)

Agenda

- ◆ Introduction
- ◆ Current State of OpenVMS Systems
- ◆ A System Management Framework for OpenVMS

OpenVMS Today – Rock Solid

- ◆ Highly Reliable, Available, Secure
- ◆ VMS is extremely resilient
- ◆ Highly disaster tolerant
- ◆ Security built in from the ground up
- ◆ Uptime often measured in years
- ◆ Hardware is equally reliable

These systems ^{almost} never go down!

Failures...

- ◆ Not what you are thinking...
 - ◆ Failure is the inability to complete an expected task when required
 - ◆ Not necessarily a system or database crash
 - ◆ Most failures are not the result of:
 - ◆ VMS operating system errors
 - ◆ Hardware errors (although storage may come close)
 - ◆ Most failures result from “small” problems
 - ◆ E.g. Application not “cleaning up”

Bet Your Business

- ◆ Most OpenVMS systems are mission or business critical
 - ◆ Application availability is essential to keeping business processes running
 - ◆ Unplanned system or application downtime is usually a severe problem
- ◆ What is the impact to your business if your VMS application is not available?

Current State

- ◆ Many systems were put into service 15-20 years ago (1990s)
- ◆ The VMS experts have moved on, retired, or are no longer available
- ◆ Systems often receive a bare minimum of updates and maintenance
- ◆ Many systems sit in a corner and run.

The background is a solid blue gradient. A wavy, horizontal line in a slightly lighter shade of blue runs across the upper portion of the slide, creating a sense of depth or a horizon line.

It's easy to become
complacent...

Complacency is Dangerous

- ◆ Has the reliability of OpenVMS made you complacent?
- ◆ Are you able to recognize and prevent a problem *before* it happens?

Complacency is Dangerous (cont)

- ◆ When something does go wrong
 - ◆ Do you have everything you need to repair, recover, restart? (You need these things BEFORE the failure)
 - ◆ Will you know for certain that everything has been restored to the prior running state?
 - ◆ Do you maintain historical data?
 - ◆ Necessary to know what the system was like before a problem occurred

Do you have Peace of Mind?

- ◆ Have you done everything you can to *prevent* a failure?
 - ◆ It's not just catastrophic incidents to worry about – small things can lead to outages
- ◆ Can you quickly *recover* from a failure?
 - ◆ Recover completely, quickly and confidently?
 - ◆ With no loss of data?

DKB1-[log files] TPN

DKB3-[log files] TPN

DKB3-[log files] TPN

To compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

to compile and

DE CLUSTER

ADVR - boot first

CIRC - second

THE DISKS REMOTE MOUNTED

DEV D

DEV /MULTI

DEVICE \$1\$DGA1S /PATH=

PGBO.5008-0SF3-

0010-BA31 /SWITCH

DEV /MULTI

/SYSTEM \$1\$DGA1S CMN-BACKUP

CMN-BACKUPS

S RUNS ON 1 NODE

R RUNS ON BOTH

PORT

PORT OFF

PORT ON

PRINTERS

7.3
CD
DESTROYED

7.4 CD
FOR BACKUPS



So – Do you have Peace of Mind?

Reactive Support

- ◆ Something breaks, then
- ◆ Heroic actions *may* result in a fix, but...
 - ◆ Higher possibility of some (or catastrophic) data loss
 - ◆ Longer recovery (down) times
 - ◆ Resolving problems under pressure never optimal
 - ◆ Solution probably not repeatable either
 - ◆ Root Cause Analysis?
- ◆ *May* make change to avoid problem again
 - ◆ Seldom is holistic approach taken to avoid all similar situations

Reactive = Too Late!

- ◆ Goals should be:
 - ◆ Avoid problems
 - ◆ Zero unplanned down time
 - ◆ Efficiently and confidently restore services
 - ◆ Prevent future occurrences
 - ◆ Confidence in running system
- ◆ Proactive = Just Right!

Agenda

- ◆ Introduction
- ◆ Current State of OpenVMS Systems
- ◆ A System Management Framework for OpenVMS

System Management Framework

- ◆ Proactive, not reactive
- ◆ Known baseline and history
- ◆ Consistent, Repeatable Processes
- ◆ Reliable Notification and Problem tracking
- ◆ Holistic Fixes
- ◆ Maintains history and audit

Establish a Baseline

- ◆ Critical to know what “normal” is
- ◆ Should keep track of:
 - ◆ “Known Good” - This is the baseline
 - ◆ “Last Known” – This is what was seen on the last look
 - ◆ Important to know if things are continuing to change
 - ◆ Ability to define “expected” values
 - ◆ Ability to define “required” values

Know Your History

- ◆ Need to keep historical data
 - ◆ Performance Data (T₄, ECP/TDC, PAWZ, RMU statistics)
 - ◆ Not just performance data
 - ◆ Disk Space
 - ◆ History of changes (prior state & time)
 - ◆ Logical names
 - ◆ Network configuration
 - ◆ Mounted volumes
 - ◆ SYSGEN parameter values

Extensible Tools

- ◆ Tools should be extensible
 - ◆ Don't write a tool to do one job – try to create a tool that can do many jobs
- ◆ Example – “State Machine”
 - ◆ Tool used to compare “Known Good”, “Last Seen” and “Current”
 - ◆ Used to monitor logical names, security settings, device parameters, etc

Monitor the System

- ◆ Use *repeatable* and *consistent* processes for monitoring the system and application
 - ◆ Monitor for changes
 - ◆ Watch trends
 - ◆ Detect errors and failures
 - ◆ Possible automatic correction
 - ◆ Capture historical data
- ◆ Proactive is NOT “periodic login” to “review logs”

Examples of Monitoring

- ◆ High file version numbers
- ◆ Excessive number of files
- ◆ New, modified or missing shared system logical names
- ◆ Startup log files
- ◆ Critical application log and event files
- ◆ Redundancy, availability of failover paths
- ◆ Changes to network configurations
- ◆ New or modified queues
- ◆ Queue status (jobs stuck, queues stopped)
- ◆ System resource utilization
- ◆ Process states
- ◆ SYSGEN Parameters
- ◆ Critical Configuration files
- ◆ And hundreds more...

Identify Problems/Incidents/Alerts

- ◆ Things that are not supposed to happen
- ◆ Things that should happen, but don't
- ◆ Changes that need to be validated
- ◆ Thresholds exceeded
- ◆ Trends – possible problems coming

Tracking events

- ♦ Monitoring and notification should integrate with a persistent Trouble Ticket System
 - ♦ Persistent does not mean “send e-mail” (aka “fire and forget”)
 - ♦ Persistent means there is an ongoing tracking and accounting of progress to completion.
 - ♦ And a historic reference
 - ♦ Full accountability thru issue resolution.

Trouble Ticketing System

- ◆ All incidents should be in a ticket
 - ◆ Automatic creation/update/closure by tools
- ◆ Where appropriate, use a checklist to validate successful completion of scheduled tasks
 - ◆ Report missing/late tasks
- ◆ Forces accountability – problems can't be forgotten or ignored
- ◆ TT system is for more than just “problems” – also tracks tasks, changes, history, provides audit trail

Notify Personnel

- ◆ Notification should be automatic – directly from the tools
- ◆ Critical incidents - use phone systems
 - ◆ Require acknowledgement, i.e. keeps calling until someone answers/acknowledges
- ◆ All incidents – automatically create a Trouble Ticket

Resolve the Issue

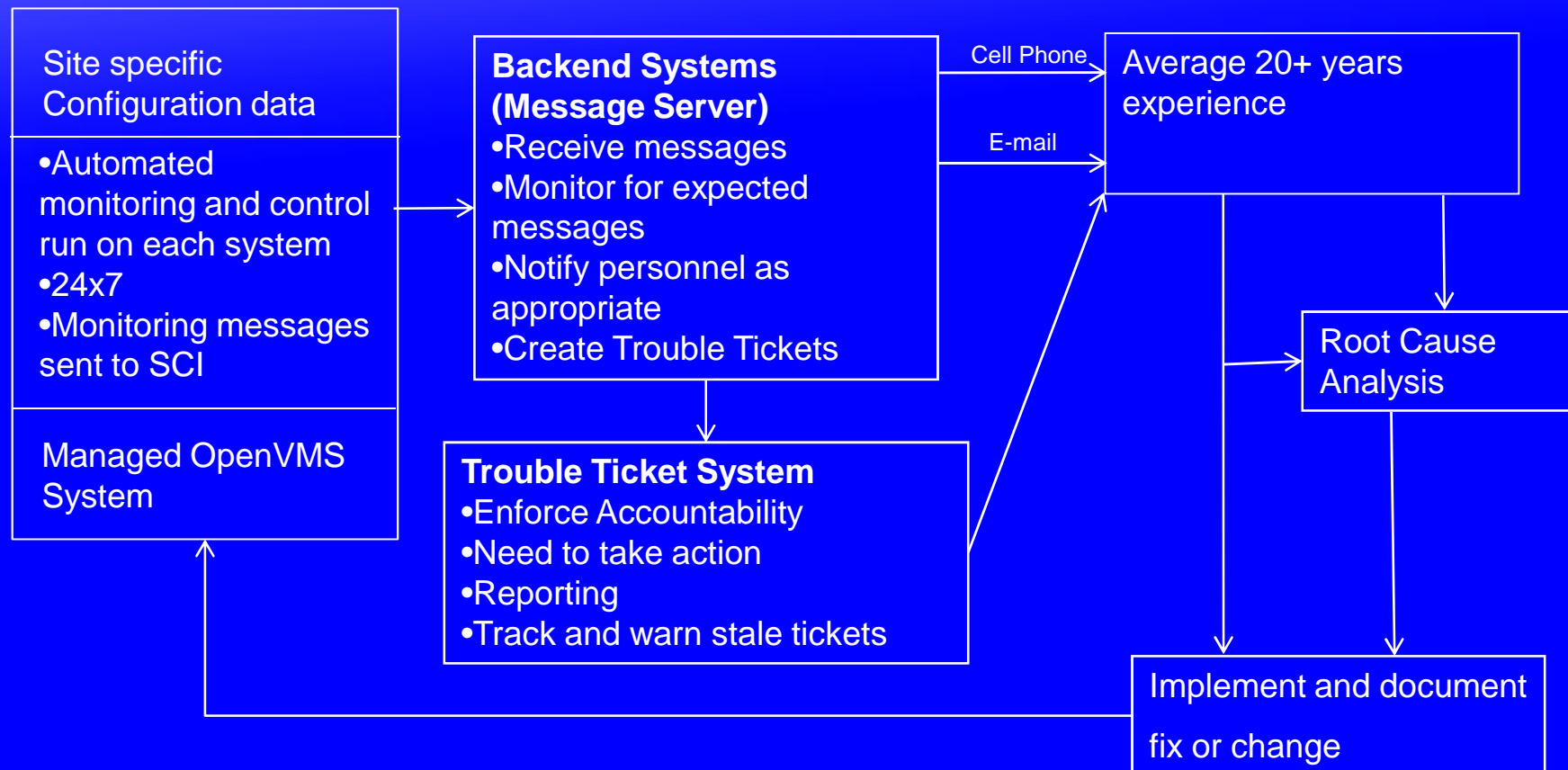
- ◆ Holistic approach
- ◆ Immediate tasks
 - ◆ Identify the problem
 - ◆ Document and Track progress
 - ◆ Fix
- ◆ Post incident task
 - ◆ Root cause analysis
 - ◆ Preventative changes
 - ◆ Enhance monitoring

The SCI Solution

Managed Systems

Support Systems

Staff



Some Closing Thoughts

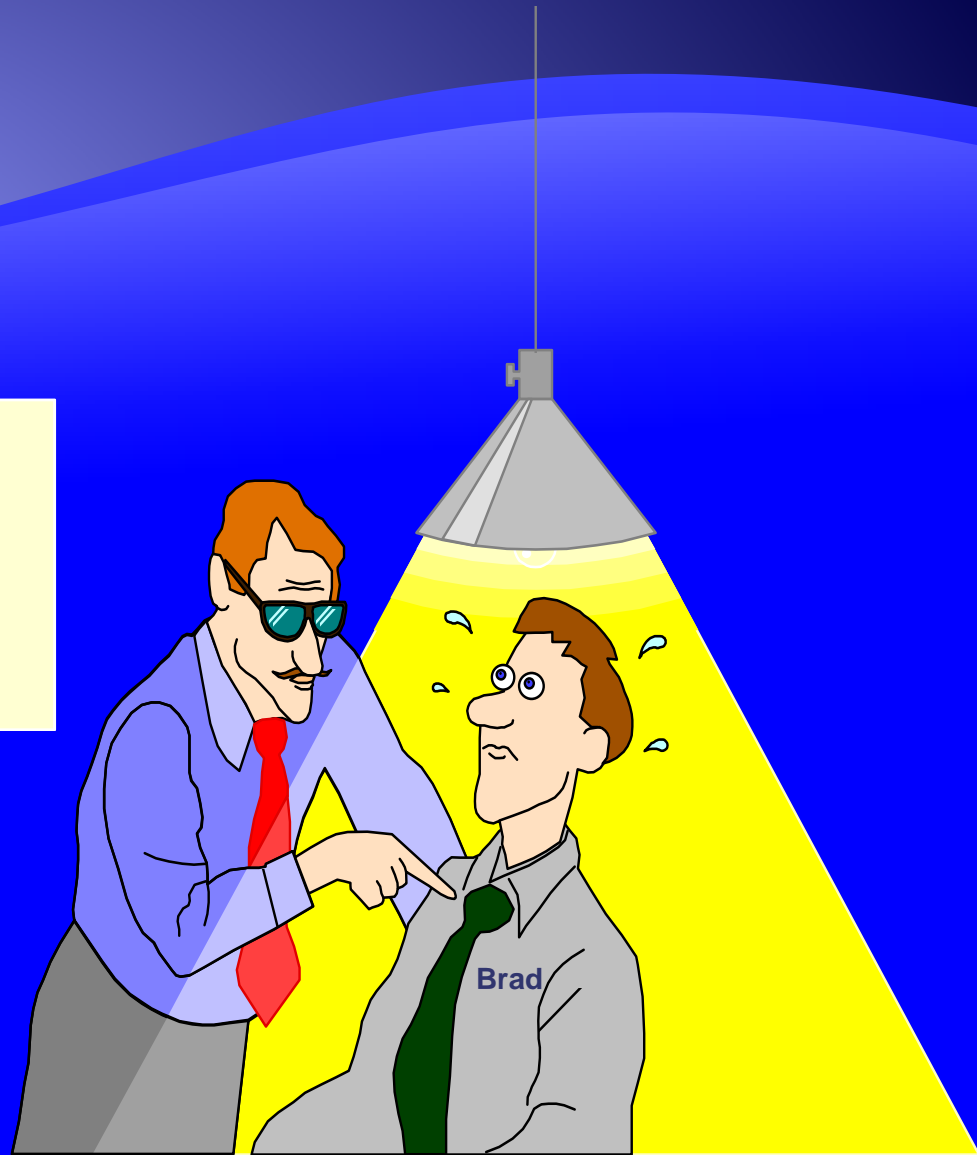
- ◆ Previous Slide:
 - ◆ This is what you should strive to build
- ◆ What business are you in?
- ◆ Focus on your business's core competencies

Questions?

Brad McCusker

mccusker@sciinc.com

www.sciinc.com



OpenVMS Remote Database Administration or System Management

- ◆ Full Service:
 - ◆ SCI is responsible for all aspects of management or administration
- ◆ Augmentation Service
 - ◆ Augment existing staff, provide 24x7 monitoring and coverage
- ◆ Solves the problem of finding, hiring and retaining quality VMS skills
- ◆ Our customers tell us we provide “Peace of Mind” for them and their VMS systems