



NonStop, Evolution through Modernisation

Shiva S

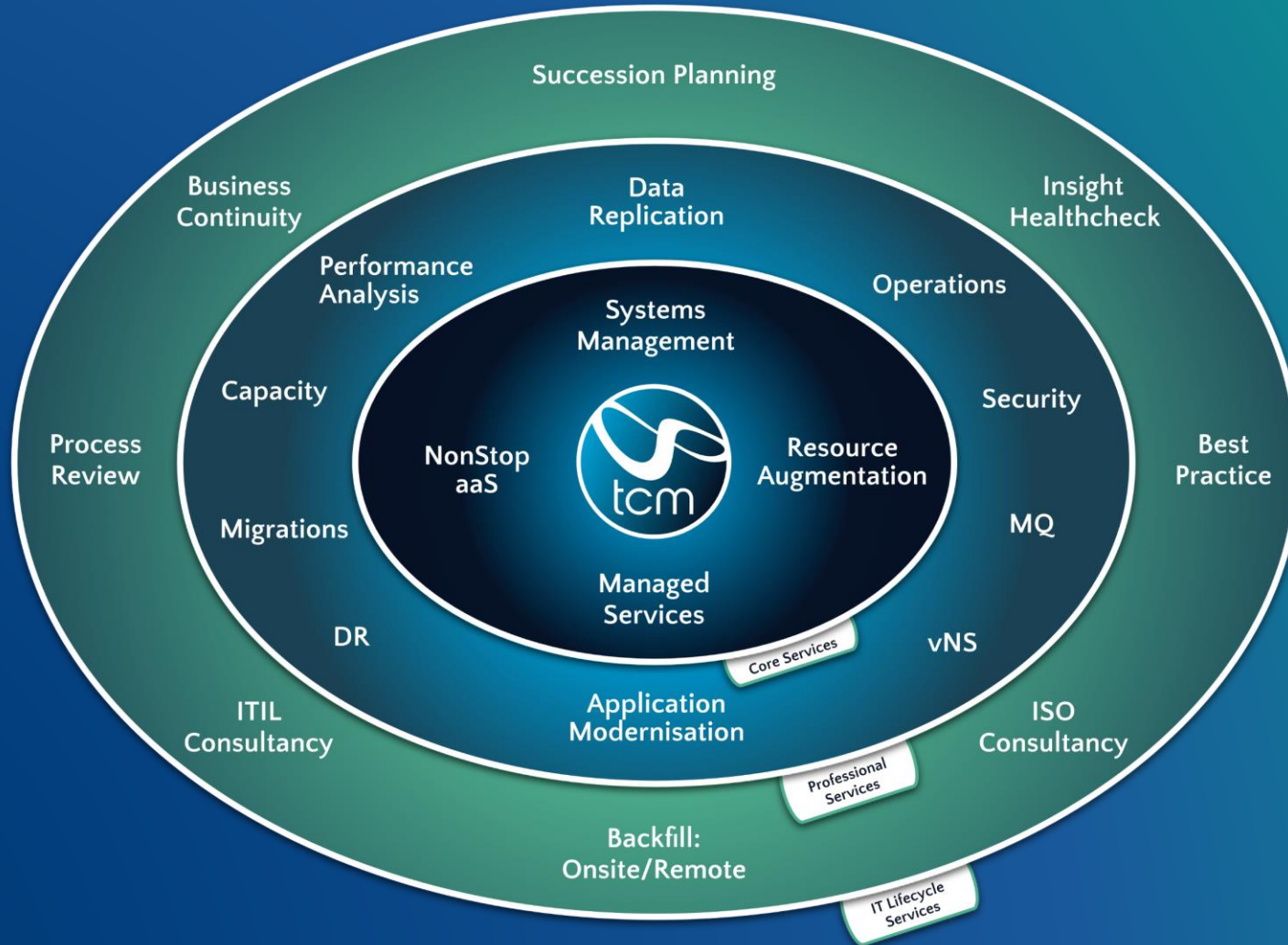


Contents

- TCM – An introduction
- Shifting Perspectives: NonStop Then v Now
- A Real-World Modernisation Example
- Live Demonstration
- Questions



The Full Suite of NonStop Services





Shifting Perspective: NonStop Then vs. NonStop Now



NonStop then

Circa 1980s



Share-Nothing architecture



Security & Compliance



Mission-Critical computing



Superior data integrity



RAID1 mirroring



Process pairs



Fault tolerant



Linear Scalability



High Availability



And more...





NonStop now To infinity & Beyond



ANSIBLE

GNU Autoconf



Everything from the past



Grafana



Git



Elastic



Jenkins



Java



Visual Studio Code



Python



Eclipse



And more...



Selenium



Gradle



DBEaver

splunk >



Confidential. Copyright TCM Solutions Ltd 2023. www.tcm.co.uk +44 (0)1592 770081



The evolution of NonStop has already begun...

— +44 (0)1592 770081 —

— www.tcm.co.uk —

— Confidential. Copyright TCM Solutions Ltd 2023. —



+44 (0)1592 770081

www.tcm.co.uk

Confidential. Copyright TCM Solutions Ltd 2023.

...and your modernisation journey...



...could be closer than you think:

Your Nonstop OS already ships with -

- Java, Python
- Make
- NSMF (Ansible setup for NonStop)
- FLOSS
- NS HTTP Server (Apache HTTP server)
- Boost Framework
- JDBC/ODBC drivers for SQL/MX

Your Organisation might already have –

- A centralised Git Repository with GitHub or GitLab
- A Jenkins master instance
- Ansible instance
- Grafana
- Elastic Stack
- Splunk
- Kafka producer setup

And on the NonStop, you can install –

- Git
- Jenkins Agent/Master
- Fortify
- Maven
- Gradle
- JMeter
- Artifactory
- Kafka libraries (to talk to Kafka producer off platform)
- Curl to communicate with -
 - Grafana / Elastic Stack / Splunk
 - Prognosis
 - Nagios
- OpenSSL & OpenSSH to communicate with -
 - Ansible
 - Jenkins
 - Git
 - Visual Studio Code
 - Nexus
 - GitHub
 - Selenium
 - Robo
 - SonarQube & more...



Ok that's great, but where do I start?

- How about a real world example?



A Real World example of:

NonStop Then vs. NonStop Now





A Real World example

Something that we can all relate to!

- In-house application using COBOL and various TACL utilities that help with day to day functions
- Development - TEDIT
- Version Control – Spreadsheets, TACL macros, RMS
- Deployment – Manual Copy, NetBatch
- Monitoring, Alerting – EMS & TACL Routines
- Setup using Enscribe and manual testing for tracking and making changes to the application

The Retro Era

- Source code changes done on TEDIT
- Comment structures tracked changes!

```
** 1.1 Begin - Updated error han
```

```
IF SEND-ERROR NOT = 0  
    PERFORM HANDLE-SEND-ERRO  
END-IF.
```

```
HANDLE-SEND-ERROR.
```

```
IF SEND-ERROR = 233  
    PERFORM SERVERCLASS-SEND  
    IF SEND-ERROR NOT = 0  
        PERFORM REPLY-WITH-E  
    ELSE
```

```
        PERFORM REPLY-WITH-P  
    END-IF  
ELSE  
    PERFORM REPLY-WITH-ERROR  
END-IF.
```

```
** 1.1 End - Updated error handling format
```

Version History			
Version	Date	Author	Amendment
1.0	2022-08-24	Shiva S	Initial version
1.1	2022-09-10	Mani Rajesh	Updated error message format
1.2	2022-10-03	Collin Yates	Improved performance of forwarding
1.3	2022-11-18	Ian Horrocks	Fixed issue causing crashes during open request handling
1.4	2023-02-02	John Groats	Enhanced error handling for specific request types
2.0	2023-05-15	Andy Vasey	Major refactor for modularity and readability

The Retro Era

- Version control done through spreadsheet for SOX compliance

Version	Date	Author	Change Description	Testing/Review	Approval	Control ID	Testing method	Test evidence	Testing results	Exception	Documentation
1	15/03/2022	Shiva S	Initial version of ROUTER code	N/A	N/A	N/A	N/A	N/A	N/A	None	N/A
1.1	10/06/2022	Mani Rajesh	Updated error handling for SOX compliance	QA Testing	Manager	CTRL_001	Manual	/test_evidence/ctrl_001	Success	None	/doc/sox_changes/ctrl_001_update.docx
1.2	05/09/2022	Ian Horrocks	Improved data validation for audit trail	Code Review	Manager	CTRL_005	Manual	/test_evidence/ctrl_005	Success	None	/doc/sox_changes/ctrl_005_improvement.docx
1.3	20/12/2022	Collin Yates	Added logging for critical operations	Audit Review	Auditor	CTRL_012	Manual	/test_evidence/ctrl_012	Success	None	/doc/sox_changes/ctrl_012_logging.docx
1.4	05/04/2023	John Groats	Enhanced documentation for control validation	QA Testing	Manager	CTRL_007	Manual	/test_evidence/ctrl_007	Success	None	/doc/sox_changes/ctrl_007_documentation.docx
1.5	18/07/2023	Andy Vasey	Security patch: Input sanitization	Code Review	Manager	CTRL_003	Manual	/test_evidence/ctrl_003	Success	None	/doc/sox_changes/ctrl_003_security_patch.docx

- While some of us used ‘intricate’ Guardian tools like RMS (while others descended to TAACL routines to keep the versions from spiralling out of control)

```
TESTCMP2
TESTCOMP
$RMP:\APPL\RMSTEST 6> icv testcmp2
COMPONENT: \APPL\RMSTEST\TESTCMP2
  1.001 26FEB2014(16:55:55) \72.LODGE.MGR
    STATE: LATEST
  1.000 11NOV2008(14:05:44) \71.SUPER.LODGE
    STATE: PRODEXEC
$RMP:\APPL\RMSTEST 7> ic testcmp2
PATH                MODE                TYPE/LANG  FILE/DEFINE
-----
\APPL\RMSTEST\TESTCMP2          SOURCE      \TANP.$RMS.RMDDL.TESTCMP2
                                TAACL
$RMP:\APPL\RMSTEST 8> add comp testcmp4,type source,lang tacl,file \tanp.$rms.
ddl.testcmp4,define =rmstest^testcmp4
COMPONENT \APPL\RMSTEST\TESTCMP4 ADDED
$RMP:\APPL\RMSTEST 9>
```

The Retro Era

- Ran obey files or compile statements manually to compile objects

```
COBOL85 /IN ROUTER, OUT $$.#ROUT.COMP, NOWAIT/
```

```
$DATA02 APPSRC 5> obey routerc  
COBOL85 /IN ROUTER, OUT $$.#ROUT.COMP, NOWAIT/  
$DATA02 APPSRC 6>  
ABENDED: $Z37C  
CPU time: 0:00:00.003  
3: Premature process termination with fatal errors or diagnostics  
$DATA02 APPSRC 6>
```

- Test cases involved manual verification of generic tests and some specific tests for the respective code change
- Production implementation was a carefully documented manual process that involved FUP DUP across systems using operational user IDs for security purposes

```
$DATA02 APPSRC 8> fup dup prodsrc.router, backsrc.router, purge  
FILES DUPLICATED: 1  
$DATA02 APPSRC 9> fup dup prodsrc.router, backsrc.router, purge  
FILES DUPLICATED: 1  
$DATA02 APPSRC 10> █
```

The Retro Era

- Used ENSCRIBE files to store and retrieve information and ENFORM to query the database

```
$DATA02 ENFORMS 59> fup copy app
param from I9;
!Title "ENFORM DATA QUERY OUTPUT"
?assign APPDATA to $data02.APPFILES.APPDATA
open APPDATA
list by app-count noprint
by loadid noprint
  loadid as A8 nohead, tab 10
!   heading "time", tab 10
  app-num as i2 nohead
!   heading "app/num/a" as I2
!   tab 15
  ((home-trans * 1.00)/(delta-time / 1000000)) as f6.2 nohead
!   heading "home/trans" as F6.2
!   tab 23
```

- Alerting and monitoring in production involved scrolling through EMS events or polling for messages in EMS through TAACL macros

```
$SYSTEM SYSTEM 28> emsd 22:30
23-09-10 22:34:00 \TCMVNS.$X7RV      TANDEM.SPI.0      000001 This is a
non-critical message
23-09-10 22:34:18 \TCMVNS.$X7RV      TANDEM.125.0      000001 This is a
critical message
23-09-10 22:35:54 \TCMVNS.$X7RV      TANDEM.STORAGE.0  000001 Error in
CONFIG file; Retcode ?; ErrDetail ?
```


The Modern Trajectory

- On the NonStop
 - Git
 - Jenkins
- On Windows servers / Linux distributions
 - Elastic / Grafana
 - Ansible
- On user machines
 - Visual Studio Code / Eclipse IDE
 - GitHub Desktop / IDE Git plugin
 - Postman
 - DBeaver





Demonstration

Using Git, Jenkins, Elastic, DBeaver & Postman





A Real World example

Then:

- Development - TEDIT
- Version Control – Spreadsheets, TACL macros, RMS
- Deployment – Manual Copy, NetBatch
- Monitoring, Alerting – EMS & TACL Routines
- Setup using Enscribe and manual testing for tracking and making changes to the application

A Real World example

Then:

- Development - TEDIT
- Version Control – Spreadsheets, TACL macros, RMS
- Deployment – Manual Copy, NetBatch
- Monitoring, Alerting – EMS & TACL Routines
- Setup using Enscribe and manual testing for tracking and making changes to the application

Now:

- Development – Visual Studio Code
- Version Control - Git
- Deployment – Jenkins
- Monitoring, Alerting – Elastic
- Using Dbeaver to manage your SQL database and PostMan for testing and other automation testing tools



A Real World example

Why did we take you through this?

- The Benefits

Rewards for the picking

- ✓ Lower costs
- ✓ Reduce risks
- ✓ Fast feedback
- ✓ Faster delivery
- ✓ Consistent and stable system
- ✓ Collaboration improves business ability





A Real World example

Why did we take you through this?

- The Benefits
- But also, this is just one of many small steps we can take on a journey to modernisation.
- You need to find yours...



Your journey

Finding out what works for you

- No production impact
- Easy install

Git

- Automate builds
- Create dependencies

Make

- Trigger builds
- Setup notifications

Jenkins

- No production impact
- Monitor everything

ELK Stack

- Automate tests
- Create code coverage

Boost/Selenium

- Trigger tests
- Setup notifications

Jenkins

C, C++, Java

Language set

API Gateway

Rest Interface

VS Code, Eclipse

10K to 1M TPS

Kafka

Modern IDEs

Ansible Orchestration

SQL/MX

DBS for NS & others

Deployment automation



A Real World example

Why did we take you through this?

- The Benefits
- But also, this is just one of many small steps we can take on a journey to modernisation.
- You need to find yours...
- ...here a few bonus tips to get you started



HPE & Vendor products that help you modernize (by minimizing your effort)

Speak to
TCM



- **HPE NonStop HTTP Server** – NonStop deep port of Apache’s famous HTTP server that runs about 25% of all websites
- **HPE NonStop Manageability Framework (NSMF)** with out of the box frameworks to support your Ansible modernization effort
- **NonStop In-Memory Cache (NSIMC)** which is a port of REDIS
- **HPE Java Transaction & Persistence APIs** – Provide NonStop integrity to your existing Java code without having to do any coding!
- **HPE API Gateway** – Bring Rest and other API standards to NonStop
- **HPE Pathsockets, NSJI, NSFTI**
- **Nexbridge NSGIT** – Guardian & ENSCRIBE support with familiar Guardian use and feel for Git
- **Comforte EscortSQL/ Tandsoft E2SQL** – remove Enscribe files without code changes and start using NonStop SQL
- **Comforte Jpath** – remove Screen COBOL green screens without code changes and start using HTML web GUI interfaces



HPE & Vendor products that help you modernize (by minimizing your effort)

Speak to
TCM



Summing up...

before we close

- ✓ Don't let the jargons put you off!
- ✓ Innovations & outcomes without technical debt
- ✓ Clarity better than certainty
- ✓ Micro steps towards MicroServices
- ✓ Era of Phoenix Servers
- ✓ Collaboration improves business ability





Thank You



Questions?



TCM – The NonStop Experts

www.tcm.co.uk