

# HDT-4470: Automating Automation, Cheaply

Michael Freeman, Sr. Applications Development Analyst

- [mlfreeman@gmail.com](mailto:mlfreeman@gmail.com)
- [michael.freeman@bkfs.com](mailto:michael.freeman@bkfs.com)

Andrew Detandt, Systems Manager

- [andy.detandt@bkfs.com](mailto:andy.detandt@bkfs.com)

## InterConnect 2017



# Session Overview

- Who Are We
- History
- Hardware
- Software
- Lessons Learned
- Questions

## Who is BKFS?

- Based in Jacksonville, FL
- Formerly known as Lender Processing Services until 2014
- Largest single provider of technology, services, data, and analytics to the U.S. mortgage industry.



# History

- 2009
  - Recruited to help start automated testing team for the Servicing Technologies division.
  - Decided to use RFT after evaluating competition
  - Lab would have to run multiple versions of Windows and IE
    - XP + IE 7, XP + IE 8, Win 7 + IE8, Win 7 + IE9, etc.
    - Only viable option: virtual machines.

# History

- 2010 – First lab: XenServer on 10 desktops
  - Free, worked, but with limits
    - Performance wasn't the best
    - Thin provisioning of VMs wasn't available
- 2011 – Second lab: VMware ESXi Free Edition – same machines
  - Thin provisioning enabled us to put more VMs in.
  - Free version could only be managed manually.
- 2011/2012 – Third lab: Archipel VM orchestrator + Fedora 15
  - Free, polished UI, but the UI was slow and the backend had issues
  - Gave us ideas for later though

## How was all this managed?

- Perl
- More Perl
- Manual Labor
  - Major tasks (e.g. new RFT version)
  - Deciding which scripts run on which machines

Is this painful?  
Yes!

## Better Way

- Our initial vision was simple: "Automate Automation"
  - Try to automate as many of our tasks as we could so we could just keep coding.
- We saw what you could do with cloud companies.
  - Cost and security restrictions ruled out actually using them.
- Why didn't we use an existing open source VM package?
  - Complexity of setup
  - Automation doesn't need a lot of what they offer

# The Present Lab

- Multiple levels of labs
  - "Main" lab: 168 possible VMs up across 14 hosts
    - Hosts all have 32 GB RAM and SSDs (256/512 GB)
    - All automated testing for other groups goes on here
  - "Dev" labs
    - Older hardware (previous main-lab machines)
    - Team tests scripts on here before they go live
- All lab virtual machines use standardized Windows images

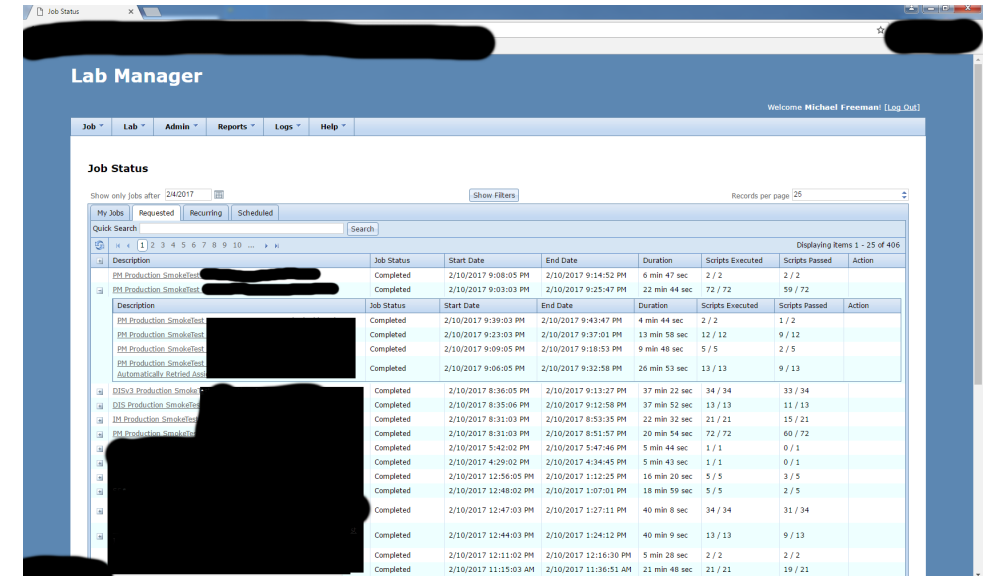


# Lab Software

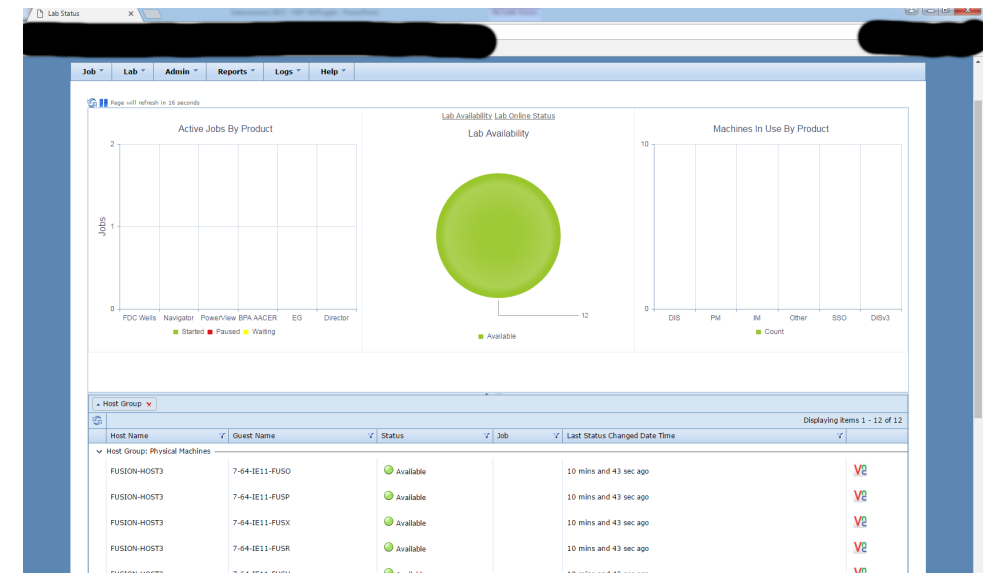
- Created In-House
  - Lab Manager
    - Logging Framework
    - RQM Bridge
  - Virtual Machine Manager
- Not Created In House
  - RFT
  - RQM
  - Linux/QEMU/KVM/libvirt
  - Windows

# Lab Manager

- Started on it in 2011/2012
- First one of our current tools
- Simple goal:  
Automate script execution
- Perl scripts were limited
  - Required manual load balancing of scripts
  - Required coordination among team members



Description	Job Status	Start Date	End Date	Duration	Scripts Executed	Scripts Passed	Action
PHT Production SmokeTest	Completed	2/10/2017 9:08:05 PM	2/10/2017 9:14:52 PM	6 min 47 sec	2 / 2	2 / 2	
PHT Production SmokeTest	Completed	2/10/2017 9:03:03 PM	2/10/2017 9:25:47 PM	22 min 44 sec	72 / 72	59 / 72	
PHT Production SmokeTest	Completed	2/10/2017 9:39:03 PM	2/10/2017 9:43:47 PM	4 min 44 sec	2 / 2	1 / 2	
PHT Production SmokeTest	Completed	2/10/2017 9:23:03 PM	2/10/2017 9:37:01 PM	13 min 58 sec	12 / 12	9 / 12	
PHT Production SmokeTest	Completed	2/10/2017 9:09:05 PM	2/10/2017 9:18:53 PM	9 min 48 sec	5 / 5	2 / 5	
PHT Production SmokeTest	Completed	2/10/2017 9:06:05 PM	2/10/2017 9:32:58 PM	26 min 53 sec	13 / 13	9 / 13	
DISK2 Production SmokeTest	Completed	2/10/2017 8:36:05 PM	2/10/2017 9:13:27 PM	37 min 22 sec	34 / 34	33 / 34	
DISK2 Production SmokeTest	Completed	2/10/2017 8:35:06 PM	2/10/2017 9:12:58 PM	37 min 52 sec	13 / 13	11 / 13	
PHT Production SmokeTest	Completed	2/10/2017 8:31:03 PM	2/10/2017 8:53:35 PM	22 min 32 sec	21 / 21	15 / 21	
PHT Production SmokeTest	Completed	2/10/2017 8:31:03 PM	2/10/2017 8:51:57 PM	20 min 54 sec	72 / 72	60 / 72	
PHT Production SmokeTest	Completed	2/10/2017 5:42:02 PM	2/10/2017 5:47:46 PM	5 min 44 sec	1 / 1	0 / 1	
PHT Production SmokeTest	Completed	2/10/2017 4:29:02 PM	2/10/2017 4:34:45 PM	5 min 43 sec	1 / 1	0 / 1	
PHT Production SmokeTest	Completed	2/10/2017 12:56:05 PM	2/10/2017 1:12:25 PM	16 min 20 sec	5 / 5	3 / 5	
PHT Production SmokeTest	Completed	2/10/2017 12:48:02 PM	2/10/2017 1:07:01 PM	18 min 59 sec	5 / 5	2 / 5	
PHT Production SmokeTest	Completed	2/10/2017 12:47:03 PM	2/10/2017 1:27:11 PM	40 min 8 sec	34 / 34	31 / 34	
PHT Production SmokeTest	Completed	2/10/2017 12:44:03 PM	2/10/2017 1:24:12 PM	40 min 9 sec	13 / 13	9 / 13	
PHT Production SmokeTest	Completed	2/10/2017 12:11:02 PM	2/10/2017 12:16:30 PM	5 min 28 sec	2 / 2	2 / 2	
PHT Production SmokeTest	Completed	2/10/2017 11:15:03 AM	2/10/2017 11:36:51 AM	21 min 48 sec	21 / 21	19 / 21	

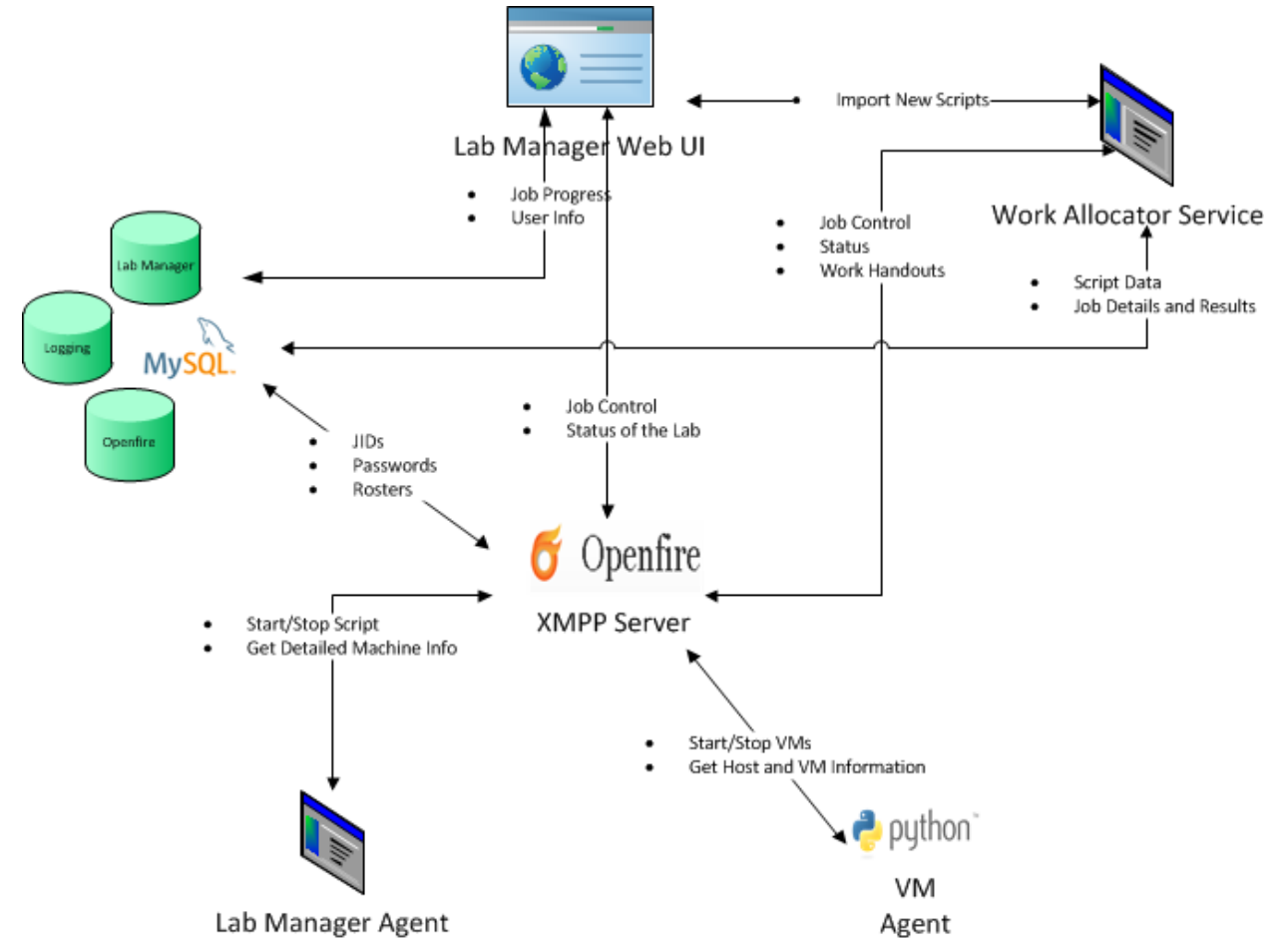


# Lab Manager

- We looked at RQM automation support but it was limited too
  - Can't share lab machine across multiple RQM projects
    - Have to manually connect to said machine and point it to other project
  - When we started, Test Automation Cells didn't exist in RQM yet.
    - You had to explicitly specify exactly which machine to send scripts to.

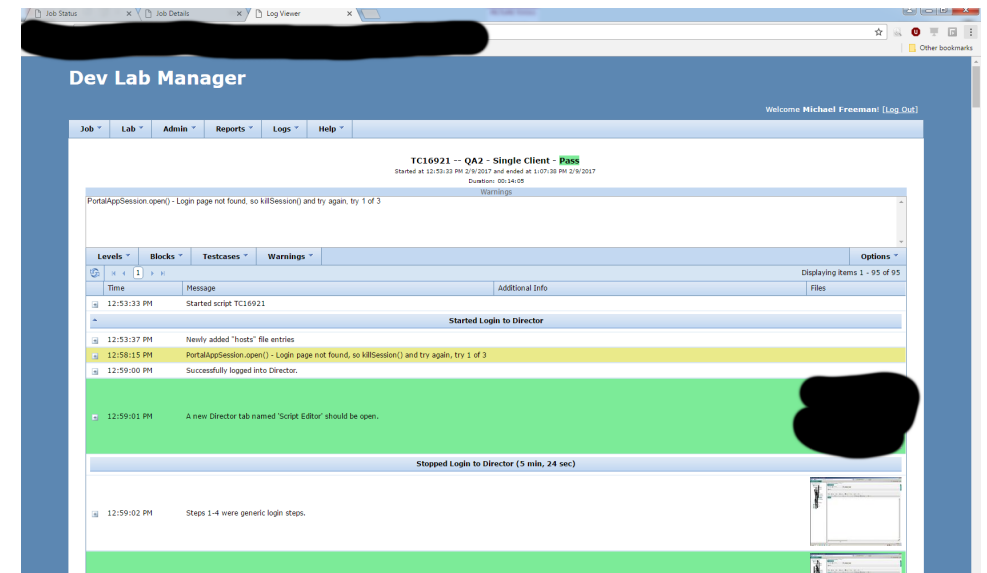
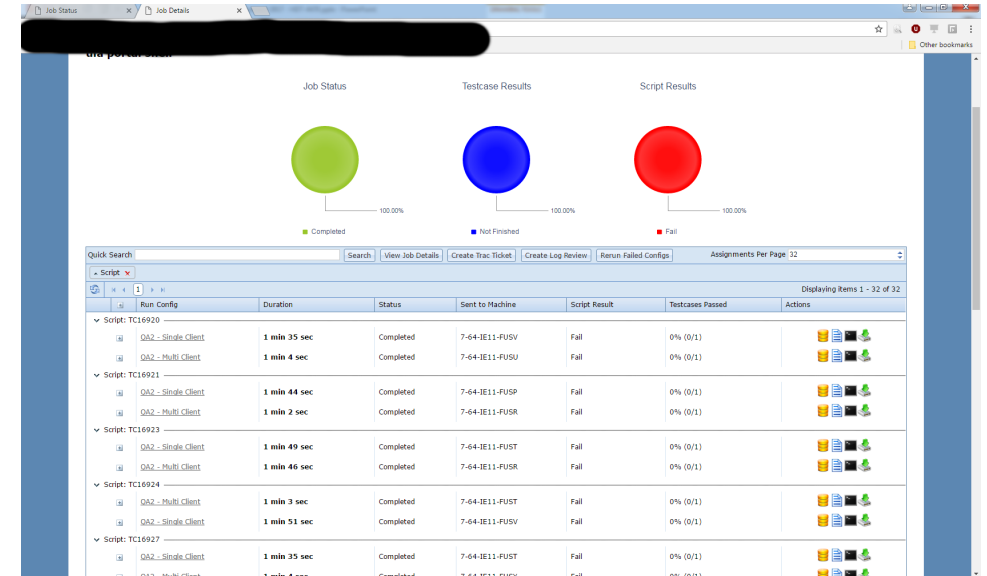
# Lab Manager

- Multi-tier system
  - Web UI
  - Service API
  - Backend Service
  - Agents on every VM and physical machine
  - Eclipse plugin
- Uses XMPP as a service bus of sorts



# Lab Manager

- Multiple execution requests can be submitted at the same time.
  - Even for different Windows/IE versions
- System monitors lab and finds open spaces to run scripts
  - Turns on/off Windows VMs based on demand.
- Common execution scenarios (e.g. regression for product X) can be saved for quick re-use.



# Lab Manager

- E-mail notifications when work requests finish.
- Service API allows tools such as UrbanCode Deploy to automatically request work
  - (e.g. Run smoke test automatically after build is deployed)



The screenshot shows the 'Lab Manager' web interface. At the top right, it says 'Welcome Michael Freeman! [Log Out]'. Below this is a navigation bar with tabs: Job, Lab, Admin, Reports, Logs, and Help. The main content area is titled 'Submit a New Job'. Under this title, there is a section 'Create from wizard' which contains three dropdown menus: 'Select a product to test' (with 'Select a product...' as the current selection), 'Against which environment' (with 'Select an environment...' as the current selection), and 'And what type of test?' (with 'Select a category...' as the current selection). A 'Next' button is located at the bottom right of the wizard section.

# Lab Manager – Overall

- Overall, everyone gets their work done faster.

Product	Type of Test	Execution Times		Time Saved
		Last release w/out lab manager	First release with lab manager	
A	Smoke	36 minutes	15 min 46 sec	20 minutes
	Regression	2 hours, 48 minutes	2 hours, 13 minutes and 15 seconds	35 minutes
B	Smoke	43 minutes	19 minutes and 18 seconds	24 minutes
	Regression	2 hours, 41 minutes	1 hrs 19 min 2 sec	1 hour, 22 minutes

- Most importantly, **automation scripts can be run and reviewed by other teams on their own.**

# Lab Manager – Logging Framework

- As part of all this, we created a custom object-based logging API
  - We need more than just free-form text describing what happened.
    - Screenshots, uploaded/downloaded/generated files, test variables, etc
  - We output to multiple formats
    - 2 forms of HTML, and one PDF layout now
    - Script writers are not allowed to think about what it looks like in the log
  - Structured objects force script writers provide a baseline level of detail
    - Free-form text APIs (e.g. log4j) can let people forget to include info
- Logging DB currently holds 2+ years of data
  - 850,000 script runs
  - 45-50 million log entries
  - 1 TB+ of distinct files (screenshots, etc) based on SHA-512 checksum



# RQM Bridge

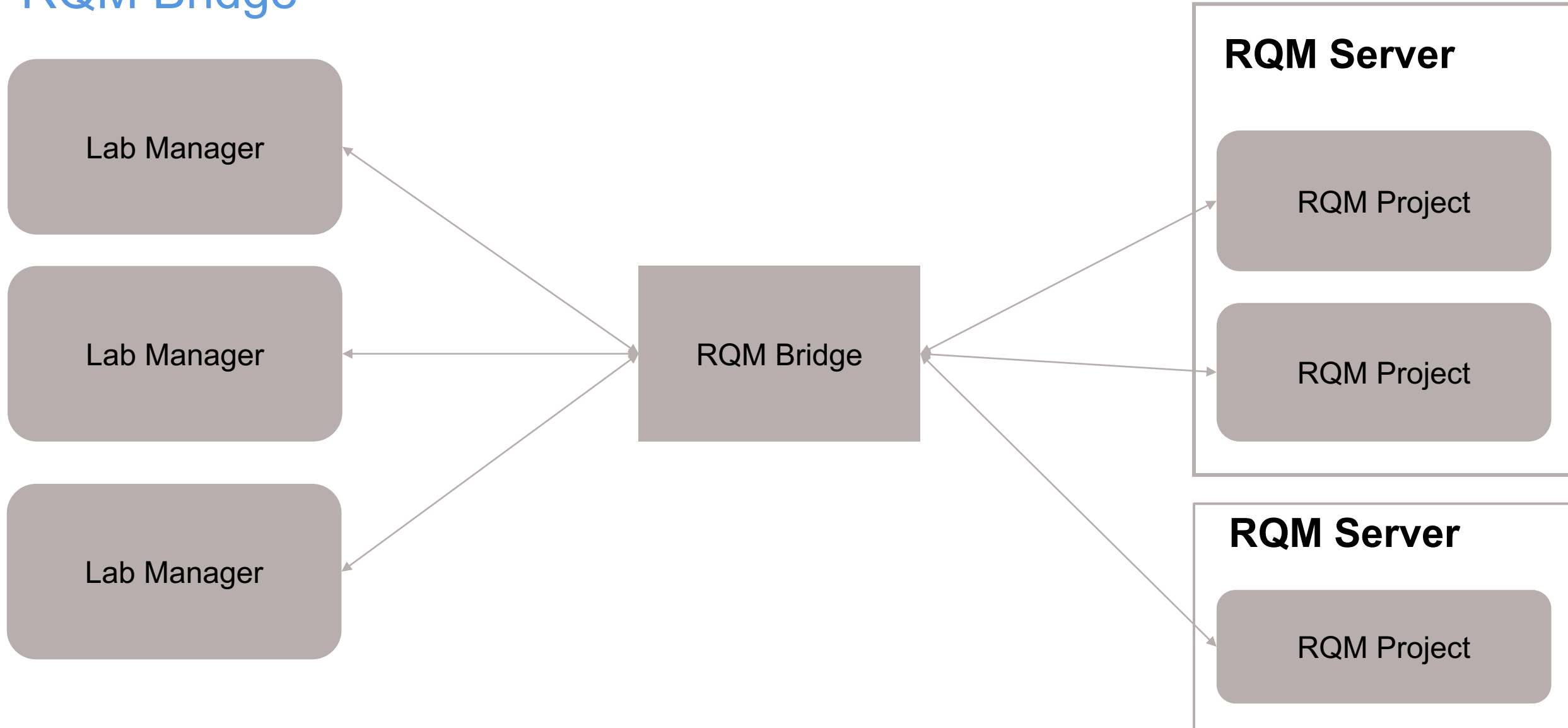
- 2014: Another automation team was merged into ours.
- Previous QA groups working with them were used to the out of the box RQM process.
- We wanted to keep as much of that as we could and still fit it in to our system.

I wrote a bridge to connect the two.

# RQM Bridge

- Java application
- Powered by OSLC4J
- Based on guide on jazz.net  
<https://jazz.net/wiki/bin/view/Main/RQMTestAutomationAdapterAPI>
- RQM Bridge can dispatch work from multiple RQM projects to one Lab Manager and intelligently track everything.

# RQM Bridge



# RQM Bridge

- Bridged the two service APIs
  - Takes requests from RQM.
  - Translates them to Lab Manager requests
  - Monitors progress in Lab Manager, updating RQM
  - Gets final log from Lab Manager Logging DB.
  - Sends result and log back to RQM as RQM-friendly HTML
- Requires us to use our own logging overlay framework in scripts.
- Could not make it work anywhere near as well with RFT's default logging API / pipeline.

# RQM Bridge

Execution Console >

❗ Execution completed. Overall verdict: **Fail**



RT\_DCT\_RTC:36496-Positional Move and Substring for numerics - client request Execution

Close

Stop Run

Show Result

Machine Name: labmanager  
IP: 192.168.1.1  
Adapter Name: Lab Manager Adapter

## Summary

Test Script Name: Auto\_QA2\_DCT\_TC21090  
Browsers: Internet Explorer 11  
Database: Oracle  
Operating System: Windows 7



100%

## Categories

## Status

❗ Test will run in the OS and browser "Windows 7 x64 IE11".

✔ Submitted job to Lab Manager as job ID 90891.

❗ If this script fails, it will be automatically re-executed by the Lab Manager before the Lab Manager reports anything to RQM.

❗ Assignment ID 2107625 started at 06:54:03 AM February 16, 2017

❗ Assignment ID 2107625 ended at 07:04:19 AM February 16, 2017

⚠ 07:04:22 AM February 16, 2017: Waiting for automatic retry to run.

❗ Assignment ID 2107625 ended at 07:04:19 AM February 16, 2017

⚠ 07:04:27 AM February 16, 2017: Waiting for automatic retry to run.

❗ Assignment ID 2107625 ended at 07:04:19 AM February 16, 2017

# RQM Bridge

Sections

Weight Distribution

Test Environment

Defects

Notes

Categories

Execution Variables

Custom Properties

Previous Results

Result Details

Formal Review

State of Test Artifacts

Associated E-Signatures

Show All Sections

History

311: DCT 1.12 & FC 1.12

Weight: 100

Result Details

07:10:27 AM February 16, 2017 Started script DCT\_TC21090

Description

Setup PositionalMove and Compute functions, run Data Conversion, and verify query returns correct data

Arguments

Index	Name	Required	Validation Passed	Value Present	Default Value Present	Default Value Used	Value	Comment
0	\$site	No	Yes	Yes	Yes	No		Optional argument \$site (Environment to run on) was supplied in at launch ("QA2") and will be used for this script.
1	\$userName	No	Yes	Yes	Yes	No		Optional argument \$userName (The user name of user) was supplied in at launch ("lbwuser1") and will be used for this script.
2	\$company	No	Yes	Yes	Yes	No		Optional argument \$company (The company name) was supplied in at launch ("loanboard") and will be used for this script.

07:10:44 AM February 16, 2017 Opened Internet Explorer 11.0.9600.18314

- URL =
- Session Process ID = 2692
- Tab Process ID = 700
- Window Handle = 524924

07:11:04 AM February 16, 2017 Desktop Login Attempt

- User =
- Password =
- Company =

22

3/22/17

# RQM Bridge - Surprises

- Custom Test Automation Script Types don't allow you to have variable arguments from instance to instance.
  - Our scripts don't always take the exact same number of parameters.
  - We have pre-defined "run configurations" in Lab Manager.
    - Each one of our custom test automation script instances just points to one of those.
    - The "run configuration" is a combination of the script and a specific set of args for it.
- The bridge had to be ready for connections to RQM being dropped a lot.
- Log returned to RQM must be simple HTML but you can include screenshots.
- RQM does *\*not\** de-duplicate file attachments (e.g. screenshots)
  - An infinite loop accidentally filled up one of our RQM servers until I made the bridge do de-duplication on its end.


# Virtual Machine Manager

- Second major tool we created
- Started life as part of Lab Manager
- Pulled it out into a separate tool so we could offer a copy to another group
  - They use it to manage their dev environment VMs.
- Currently, it shares/reuses the Linux agent from Lab Manager.
- Provides a web front end for lower-level tasks that used to be restricted to trained team members running Perl scripts



# Virtual Machine Manager

- Offers "infrastructure as a service" type capabilities
- Start/stop VMs individually or in bulk
- Monitor lab health (physical machine status, free space, etc)

 VM Manager

Provision a VM ▾ Manage Assets ▾ Logout Register Help ▾

Hello, Michael Freeman

My Factory

DEVFACTORY [REDACTED]

Code

Windows 7 x64 IE11

Windows 7 x64 IE9

Windows 7 x64 IE8

Windows 8.1 IE11




Windows 7 x86 IE8

Windows 7 x86 IE9

My Hosts


Cleanup

Drag a column header and drop it here to group by that column

Host	In Use RAM	SSH	Is Online	Uptime	Active VMs	Free Space	Include in Action
TEAM1-LAB1 [REDACTED]	8.84Gb		true	31 days, 2 hours, 54 minutes, 31 seconds	4	/ has 125.87 GB /opt has 350.78 GB	<input checked="" type="checkbox"/>
TEAM2-LAB5 [REDACTED]	10Gb		true	25 days, 3 hours, 2 minutes, 31 seconds	4	/ has 17.72 GB /opt has 6.42 GB	<input checked="" type="checkbox"/>
TEAM1-LAB2 [REDACTED]	6.82Gb		true	31 days, 2 hours, 16 minutes, 31 seconds	0	/ has 129.88 GB /opt has 64.83 GB	<input checked="" type="checkbox"/>

# Virtual Machine Manager

- Create/delete VMs on demand
  - Individually or in bulk

 VM Manager

Provision a VM ▾ Manage Assets ▾ Logout Register Help ▾

Hello, Michael Freeman

Simple Form Advanced Form

Create a Linked Clone

Basic Information

VM Name

Enter a name for the VM

Memory

Enter memory allocation in MB

Number of Virtual CPUs

Enter number of vCPUs

Host Machine

Select a host... ▾

Network Information

Type of Network Connection

Select a interface... ▾

Name of Network interface

What is the name of the network?

Network Device Driver

Select a driver... ▾

MAC Address

Fixed MAC address for the guest

Graphic Information

Display Type of Remote Console

Select a display... ▾

Remote Console Password

Enter a password

Address to Listen

Address to listen for VNC/Spice cc

Disk Information

File Name	Master Image	Storage Format	Cache Mode	Device Bus	Is Bootable?	Action
Path to some storage media to use, existing or not	Revision 226 -- ▾	Select a format... ▾	Select a cache m... ▾	Select a bus... ▾	Enter guest name	<div>− +</div>

Submit

# Virtual Machine Manager

- Manage lab images (virtual disks)
  - Track/store multiple revisions of the same image, for easy deployment and rollback

BLACK KNIGHT  
FINANCIAL SERVICES

VM Manager

Provision a VM

Manage Assets

Logout

Register

Help

Hello, Michael Freeman

Master Image Details

ID 1	Name Code	Description Disk file with automation code
Format VMDK	Initial FileName /home/administrator/VMs/code.vmdk	Limit Check-out to Latest True
Recommended RAM 2Gb	Recommended vCPUs 1	MAC Pool Required False
Bootable Image False	Hypervisor Name Prefix Code	Master Image Identifier code
OS Name Windows 7 x64 Professional SP1	Guest Login User [REDACTED]	Guest Login Password [REDACTED]

Factory Event Viewer

2/16/2017 7:27:13 AM - CHECKIN\_REVISION by [REDACTED] - Check in of Code completed successfully

2/16/2017 7:27:12 AM - CHECKSUM\_REVISION by [REDACTED] - Checksum of file, /home/administrator/VMs/code\_v226.vmdk on vmfactory.corp.finfs.net machine, was successful.

2/16/2017 7:26:54 AM - COPY\_IMAGE\_TO\_HOST by [REDACTED] - File delete was successful

2/16/2017 7:26:53 AM - UNDEFINE\_REVISION by [REDACTED] - Successfully undefined, Code\_v2404, VM but left the associated storage files intact

2/15/2017 12:36:05 PM - INITIATE\_CHECKOUT by [REDACTED] - Check out of Code completed successfully

2/15/2017 12:36:03 PM - SUPPORT\_REVISION by [REDACTED] - Successfully

Revision History

Additional Disk Drives

Current Usage

Revision History

Revision Number	Comment	Checked Out By	Checked In By	Description	File Name	Capacity	Size on Disk	Check-out Date	Check-in Date	Cloned From Revision	Action
226	sso branch 1002 ...	Nan [REDACTED]	Nan [REDACTED]	Quick3270 installed	/home/administrator/VMs/code_v226.v...	4Gb	3Gb	2/15/2017 12:3...	2/16/2017 7:27...	225	
225	SSO Branch 100...	Nan [REDACTED]	Nan [REDACTED]	Quick3270 installed	/home/administrator/VMs/code_v225.v...	4Gb	3Gb	2/14/2017 11:0...	2/15/2017 9:20...	224	
224	Quick3270 install...	Jeremy [REDACTED]	John [REDACTED]	Quick3270 installed	/home/administrator/VMs/code_v224.v...	4Gb	3Gb	2/13/2017 10:2...	2/14/2017 10:0...	223	
223	Team 2 - Sprint 2...	Jay [REDACTED]	Jay [REDACTED]	Team1 Branch 1001 Code Me...	/home/administrator/VMs/code_v223.v...	4Gb	3Gb	2/13/2017 8:03...	2/13/2017 9:42...	222	
222	SSO Merge 1002	Nan [REDACTED]	John [REDACTED]	Team1 Branch 1001 Code Me...	/home/administrator/VMs/code_v222.v...	4Gb	3Gb	2/9/2017 10:46...	2/9/2017 2:02...	221	
221	SSO Merge 1002	Saipreethi...	Danny [REDACTED]	Team1 Branch 1001 Code Me...	/home/administrator/VMs/code_v221.v...	4Gb	3Gb	2/9/2017 7:18...	2/9/2017 7:50...	220	
220	SSO Merge 1002	Nan [REDACTED]	John [REDACTED]	Team1 Branch 1001 Code Me...	/home/administrator/VMs/code_v220.v...	4Gb	3Gb	2/8/2017 11:40...	2/8/2017 2:36...	219	
219	SSO Merge 1002	Nan [REDACTED]	Kim [REDACTED]	Team1 Branch 1001 Code Me...	/home/administrator/VMs/code_v219.v...	4Gb	3Gb	2/6/2017 6:58...	2/8/2017 7:38...	218	
218	SSO Merge 1002	Nan [REDACTED]	Kim [REDACTED]	Team1 Branch 1001 Code Me...	/home/administrator/VMs/code_v218.v...	4Gb	3Gb	2/6/2017 6:57...	2/8/2017 6:58...	217	

# Rational Functional Tester

- I personally have been using it since 2007.
- The ability to treat RFT's testMain() as if it were basically a regular Java main() function made a lot of what we do possible.
  - Other automation tools aren't so easy to drive programmatically.
- It works well enough for our lab now.
  - 90% of our applications under test are web apps.
  - Other technologies:
    - .NET full applications: 2
    - 3270 terminal apps: 1
    - Java applets: 1
    - Silverlight applications: 1
    - Delphi 2010 applications: 1

# Rational Functional Tester

- Had to build a framework that worked inside it but could bypass or extend it for some cases
  - Logging (mentioned earlier)
  - Apps that needed to use MS UIAutomation underneath for testing
    - Silverlight
    - Delphi
    - Windows Dialogs (Open, Save As, Print, etc)
    - Shared not-under-test apps (e.g. Notepad++, Notepad, etc)
  - Database connectivity
  - Webservice connectivity
  - File IO (Excel, PDF, HTML parsing, CSV, ZIP, etc)

# Linux

- Lab machines use Ubuntu Linux 14.04 LTS
- We install QEMU/KVM to provide the actual hypervisor for Windows VMs.
- We use libvirt as the thin layer between our own agent and the virtual machines on each host.
- If you use boxes with solid state drives, the base Ubuntu OS layer can work basically out of the box with no real changes needed.

# Linux

- We can comfortably get about 3 VMs a core, before RFT slows down
- We never let guest memory exceed 75% of the host's total
  - Hypervisors need 10-20% additional RAM per VM for their purposes
    - (e.g. a VM with 2 GB RAM will take 2.2-2.4 GB RAM on the host)
- On a quad core box with 32 GB RAM, this gets 12 VMs with reasonable performance.
- When rolling out new virtual disks to the lab, Windows will sometimes throw fits about product activation. RFT still works even if Windows is unactivated.

# Lessons Learned

- Why did we code so much ourselves?
  - Cost of commercial tools
  - Hassle of open source tools
  - Learning experience for the team
    - Junior developers who want to go on to "real" development roles
    - Provides something to break up the routine of scripting
  - On a team of 10+ it took well under 1 person's time-equivalent to develop this over the years.
- Build everything in small modules.
  - You never know what you might realize you can create if you've created the right microservices.
- Automation doesn't need all the shinier aspects of cloudy computing
  - Create, delete, start, stop, update disks...that's all you really need to do to your VMs.



# Lessons Learned

- A lab shared across products has proven to be a better value than creating separate labs for each product, even with the cost to create our in-house tools.
  - Execution is faster. Lab machines are never left unused if there's work to be run. Not every product has to shell out for a lab of its own.
  - Automation practices are kept to a global standard so they work on a standard VM.
  - When you have a shared lab, make sure to share the automation team.
    - We spend more time talking amongst ourselves about automation problems (e.g. unusual UI controls) than we do talking with product-specific people about AUT matters.
    - If we were split by products, automation developers would end up reinventing the wheel a decent amount of the time.

## Lessons Learned

- Every tool (e.g. RQM) has its surprises that you can't see until you dive in, so when evaluating one, always quadruple any estimates you initially come up with.
- The forums on [jazz.net](http://jazz.net) were helpful when writing the RQM bridge.

# Conclusion

- In short, sophisticated test automation capabilities are much easier and cheaper to obtain than you might think!
- Hopefully, my notes will help you on your journey towards test automation excellence.

Questions?

InterConnect  
2017



Thank you for attending my talk today.

The slides are on GitHub  
<http://github.com/mlfreeman2>

InterConnect  
2017

