Today’s Lineup:

Todd DeCapua  
Executive Director  
Global Technology  
JPMorgan Chase & Co.

Petar Puskarich  
AVP  
Performance Engineering  
Bank of America

Jonathon Wright  
Digital Therapist  
Digital Assured

Stuart Moncrieff  
Web Performance Evangelist  
MyLoadTest.com
Vivit SIG Talk:
Quality & Testing
“Proven Practices”
9 January 2018
Welcome to #VivitSIGTalk
Future topics

If you are interested in participating as a SIG Talk speaker for one of our 2018 Quality and Testing topics or want to suggest a speaker or topic, please take a moment to **answer our survey**.
Today’s Speakers:

Petar Puskarich
AVP
Performance Engineering
Bank of America

Jonathon Wright
Digital Therapist
Digital Assured

Stuart Moncrieff
Web Performance Evangelist
MyLoadTest.com
Webinar Housekeeping

- This “LIVE” session is being recorded
- Recordings are available to all Vivit members
- To enlarge the presentation screen, click on the rectangle in the upper right hand corner of the Presentation pane
Webinar Control Panel

- Session Q&A:
  Please type questions in the Q&A pane and click on “Ask”
- Choose the language in which you would like to ask your questions

#VivitSIGTalk
Tweet Your Questions

#VivitSIGTalk
Proven Practice: “Performance Engineering Personnel Mining?”
Performance Engineering
Personnel Mining

What to look for:
Where to find them:
What questions to ask:
What to avoid:
What to look for:
- Good mix of soft/technical skills
- Think like an engineer!
- Ability to communicate with engineering details
- Asking questions related to interview topics
- Willingness to learn YOUR tooling/procedures
- Ability to quickly estimate engineering specifics
Performance Engineering
Personnel Mining

Where to find them:

Staff recommendations
Traditional technical staffing firms
Staffing sites: Indeed, Monster, LinkedIn, Corporate hiring portals
Contacts made through conference and industry group participation
Performance Engineering
Personnel Mining

What questions to ask:
- Open ended questions that show thought process
- Questions pertaining to “I” not “WE” are important
- YOUR tooling specific questions – easy and hard
- Performance related modeling questions
- Explanation of how they became a PE
- Validation questions of shared history
- Listen to the candidate – Push back on questions
- Questions from the candidate
Performance Engineering Personnel Mining

What to avoid:
- Resume Fraud
- Technical Interview Fraud
- Candidate Fraud
Proven Practice:
“What are Digital Experiences (DX) within the Digital Performance Lifecycle?”
What is Digital Experiences (DX) within the Digital Performance Lifecycle
In this session, Jonathon will define what is Digital Experiences (DX) and why they are so important within the Digital Performance Lifecycle (DPL), and proven practices to help your organization adopt Minimum Viable Performance (MVPx), Performance as Code (PaC) combined with Test Data as Code (TDaC) to supercharge the Continuous Performance (CPx) delivery pipeline using bots (ChatOps) to deliver higher quality software at an ever-greater pace.

Digital business systems need to be extremely responsive to change, be resilient in the face of unpredictable failure modes across machine to machine (M2M) and Internet of Things (IoT) connections and microservice endpoints.
Digital Performance (DP)

- Virtual Personal Assistants (VPA)
- Smart Advisors
- Natural language processing
- Situation Awareness
- People-Literate Technology
- Digital Experiences (DX)
- Internet of Everything (IoE)
- Human Augmentation
- Ambient Experiences
- Gesture Control
- Brain Computer Interface (BCI)
- Emotion Detection
- Head-Mounted Displays
- Virtual Worlds
- Context Brokering Platforms
- Digital Offers
- City Data Exchange
- Complex Event Processing
- Atomic Level Mass Personalization
- Connected Vehicle
- Autonomous Vehicles (C2X)
- Vehicles to Infrastructure (I2X)
- Mood Recognition

- Ultra-Intelligence (Singularity)
- Artificial Intelligence Foundations
- Deep Reinforcement Learning
- Learning / Selfware Software
- Neural Networks / Conversational
- Quantum / Fog / Edge Computing
- Cognitive Reckoning / Abstract Thinking
- Humanoid robots
- Vision / Sensors / Knowledge Replication
- Neuromorphic Hardware
- Digital Twins / Hyperconverged
- Cyber-physical systems
- Cryptocurrency Lawful hacking
- Cyber Threat Prediction / Zerotrust security
- Location-based authentication
- Growth / Performance hacking
- Flexible displays / Screenless interface
- Smart Dust / Workspace / Robots
- Nanotube Electronics
- Accumulated Reality
- Self-healing & aware systems
- Digital Out-of-Home (DOOH)
- Additive manufacturing

- Virtual / Augmented / Mixed Reality
- Immersive / Human Augmentation
- Intelligent / Enterprise of Things (EoT)
- Blockchain / Event Driven
- Mobilegeddon / Intelligent Apps
- Device Mesh / Sensorization
- Data visualization / stewardship
- Business / Digital transformation
- Tri-Modal / Shadow IT
- Interactive digital signage / Ambient
- Omni channel engagements
- Adaptive risk, trust & learning
- Microcontainerization [Unikernels]
- BI / Big Data [NoSQL]
- Wearables / Haptics / Gesture tech
- 4D / 3D printing
- Uberfication of services
- Object based storage
- Smart building technologies
- Cloud aggregator / broker
- Chaos Monkey
- Gamification
- Commercial drones (UAVs)
Digital Performance Lifecycle
Minimum Viable Performance

✓ Continuous Performance (CPx)
✓ Performance as Code (PaC)
✓ Test Data as Code (TDaC)
  - All Controlled using business terms inside a data file
  - Behind the scenes there is a very powerful data creation engine
  - Model transactions with different numbers of related rows spread across multiple tables and look data up from databases
✓ Enterprise Collaboration (ChatOps)
  - Control LoadRunner from inside Slack using VIP
  - Expose all the parameters, attributes and options as parameters
ChatOps - Demo

Visual Integration Processor

A Rapid Application Assembly Framework
Cognitive Learning

https://youtu.be/DqUZuzMBR30
Proven Practice:
“What is the next step after Version Control?”
What is the next step after Version Control?
Attention Performance Testers!

Moving your script development practices into the late 20th Century...

Let's advance to here

1970s  1980s  1990s  2000s  2010s  2020s
## A Brief history of Version Control

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1972</td>
<td>Source Code Control System (SCCS) developed at Bell Labs</td>
</tr>
<tr>
<td>1982</td>
<td>Revision Control System (RCS) developed at Purdue University</td>
</tr>
<tr>
<td>1990</td>
<td>Concurrent Versions System (CVS)</td>
</tr>
<tr>
<td>1992</td>
<td>Rational ClearCase</td>
</tr>
<tr>
<td>1994</td>
<td>Microsoft Visual SourceSafe</td>
</tr>
<tr>
<td>2000</td>
<td>Subversion (SVN)</td>
</tr>
<tr>
<td>2005</td>
<td>Git released by Linus Torvalds</td>
</tr>
</tbody>
</table>
# Version Control Integration for Load Testing tools

## IDEs for Perf Testers

<table>
<thead>
<tr>
<th>Load Testing Tool</th>
<th>Year of Integration</th>
<th>SCM</th>
</tr>
</thead>
<tbody>
<tr>
<td>LoadRunner</td>
<td>2016 (v12.53)</td>
<td>Git</td>
</tr>
<tr>
<td>NeoLoad</td>
<td>2017 (v6.2)</td>
<td>Git</td>
</tr>
<tr>
<td>SilkPerformer</td>
<td>2012 (v9.5) [earliest reference found]</td>
<td>StarTeam, Microsoft Visual SourceSafe, and PVCS</td>
</tr>
</tbody>
</table>

Performance testers are very late adopters.
What the Dev team thinks of you when you say “we don’t use version control for our scripts”...
What else can we improve?

What other development practices have we been missing out on?
## A Brief History of Static Code Analysis

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
</tr>
</thead>
</table>
| 1979 | Lint released with Version 7 Unix by Bell Labs.  
• Checks C source code without running the program.  
• Checks for things like: non-portable code, variables being used before being set, buffer overflows, etc. |
| Today | • Static Analysis tools exist for every common language  
• Built into some IDEs, or added to CI toolchain.  
• Different tools check for:  
  • Code style (e.g. StyleCop for Visual Studio)  
  • Secure programming, logic errors, race conditions  
  • Domain-specific problems |
Example: Fortify Static Code Analysis (by Micro Focus)

Scan source code written in developers' favorite programming languages.
Fortify supports a growing list of 25+ industry-leading programming languages.
But we already do manual code reviews!

- Step 1: Define internal scripting standards and a list of common scripting errors
- Step 2: **Automate** anything that is machine-checkable!
- Save your humans for doing things that are hard to automate
- Enjoy the benefits of having scripting errors highlighted in real-time, instead of waiting for a senior team member to have a look
Examples of Machine-Checkable scripting guidelines

<table>
<thead>
<tr>
<th>Scripting Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Every HTML page or JSON payload that is returned by the server should have an explicit verification check.</td>
</tr>
<tr>
<td>2. Scripts should have a list of error messages that are searched for automatically in every response.</td>
</tr>
<tr>
<td>3. No web requests should appear outside of a transaction.</td>
</tr>
<tr>
<td>4. Scripts should use the following runtime settings...</td>
</tr>
</tbody>
</table>
Who might benefit?
A Manager of Testing Services at a large bank

• Bank has paid millions of dollars for Performance Center, and has outsourced performance testing to a low-cost “testing services partner”
• Manager has low confidence in outsourcing company but has minimal staff time available for governance activities.
• Static Analysis tool monitors scripts in the Performance Center repository, and alerts when they are bad. The next contract with the outsourcer includes scripting quality SLAs.
A Practice Manager at a Consulting/Outsourcing firm

- Will be blamed if their outsourcing company loses contracts for poor quality of delivery
- But has been given a team of mostly grads, with high turnover of experienced staff
- **Static Analysis tool helps to boost the quality of the scripts the team delivers, and reduces the impact of staff turnover by ensuring consistent scripting standards.**
### A Perf Testing newbie, or a Perf Testing expert

<table>
<thead>
<tr>
<th>Beginner</th>
<th>Expert</th>
</tr>
</thead>
<tbody>
<tr>
<td>A safety net that catches a lot of the common mistakes that all beginners make.</td>
<td>Catches a lot of the common mistakes that everyone makes (because everyone makes mistakes).</td>
</tr>
<tr>
<td>Less need for support from other team members, because the static analysis tool tells them what they’ve done wrong.</td>
<td>Helps all team members be consistent with scripting practices. No more “who’s the idiot wrote this?”</td>
</tr>
<tr>
<td>Knowing that your scripts meet team standards</td>
<td>Knowing everyone’s scripts meet team standards</td>
</tr>
</tbody>
</table>
Any Perf Tester who uses version control

- Is already following one best practice by storing load testing scripts in a Git repository, but still wants to improve their workflow.
- Knows that they can use a Git pre-commit hook to automatically run their Static Analysis tool before a change can be committed to their repository.
- Improves collection of scripts in repository by incrementally adding new checks.
Upcoming Vivit Webinars

January 25, 2018
Next Generation Network Management from Micro Focus
9:00 - 10:00 AM PDT (Los Angeles), 12:00 PM - 1:00 PM EDT (New York), 18:00 - 19:00 CET (Frankfurt)
Register at:  http://www.vivit-worldwide.org/events/EventDetails.aspx?id=1051646&group=

January 30, 2018
DevOps SIG Webinar
8:00 - 9:00 AM PDT (Los Angeles), 11:00 AM - 12:00 PM EDT (New York), 17:00 - 18:00 CET (Frankfurt)
Register at:  http://www.vivit-worldwide.org/events/EventDetails.aspx?id=1048753&group=
Vivit SIG Talk

• Complete the short survey
• For more information, go to:

http://www.vivit-worldwide.org/?page=Quality_Testing

#VivitSIGTalk
Thank You
vivit-worldwide.org