Performance Engineering: New and Conflicting Trends

Alex Podelko
Sr. Performance Engineer
Amazon Web Services

1

Alex Podelko

- Specializes in performance since 1997
- Senior Performance Engineer at AWS – Amazon Aurora
  - Before worked for MongoDB, Oracle/Hyperion, Intel, and Aetna
- SPEC RG Steering Committee Member

Disclaimer: The views expressed here are my personal views only and do not necessarily represent those of my current or previous employers. All brands and trademarks mentioned are the property of their owners.

2

Performance Topics Getting More Attention

3

Algorithmic Complexity

- Time Complexity
- Space Complexity
- Big-O notation

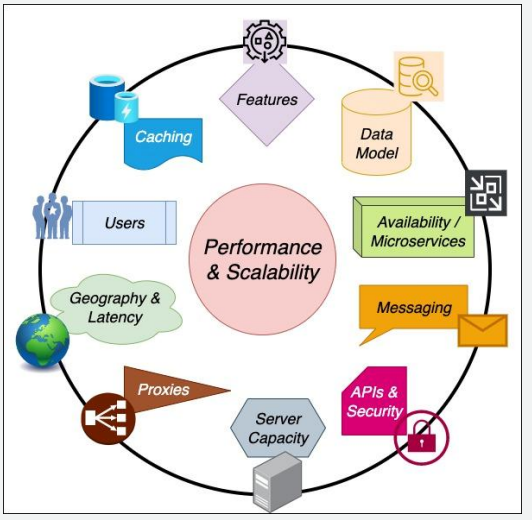
Almost in every interview around the globe !

- Connect it with practical performance engineering?

4

System Design Interview Cheat Sheet
by Vahid Dejwakh

Just ***an example*** of the changing attitude



5

AWS Well-Architected Framework

The 6 Pillars of the AWS Well-Architected Framework

- Operational Excellence
- Security
- **Reliability**
- **Performance Efficiency**
- **Cost Optimization**
- Sustainability

6

Well-Architected Trend Across the Industry

- Azure Well-Architected Framework
  - Pillars: Performance Efficiency, Cost Optimization, etc.
- Google Cloud Well-Architected Framework
  - Pillars: Performance Optimization, Cost Optimization, etc.
- Well-Architected Framework for Oracle Cloud Infrastru
  - Pillars: Performance and Cost Optimization, etc.

7

Adjusting Performance Engineering to Major Industry Trends

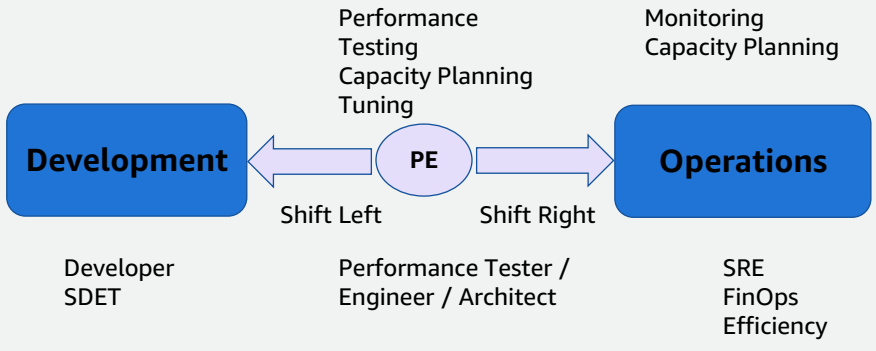
8

What Changed?

- Web
  - Centralization
  - Open / unlimited workload
- Cloud
  - Further centralization [chargeback – direct cost to performance correlation]
  - Dynamic configurations / Self-Management
- Agile / iterative development
  - Continuous Integration / Delivery [fuzzier line between Dev and Ops]

[The Past, Present, and Future of Performance Engineering]

Integrating Performance Engineering into DevOps



Expand or be Squeezed Out ?

11

Performance Engineering as an Independent Discipline

- Not well defined, no agreed Body of Knowledge, no certifications
- Fractured
  - Performance Testing
  - Web Performance
  - Capacity Planning
  - Etc.
- No significant developments recently [comparing to the increased attention]
  - Most developments are in integrated areas

12

Decreased Attention - Queuing Models

- An important approach, but not the foundation [anymore]
- Multi-core servers, horizontal scalability, and not using servers to 100% probably allow using linear models in many cases
- The hard-core queuing theory probably belongs to graduate studies (where it landed anyway)
- "Black Box" ML models are not a replacement

The Forgotten Art of Performance Modeling

Decreased Attention - Load Testing Tools

- Performance engineering shifted to
  - Other ways to mitigate performance risk
  - More closely integrated continuous performance testing
    - Proprietary solutions
- Never got track in academia anyway
  - Traditional record / playback approach (with correlation and parameterization) has some didactic value

DevPerfOps Foundation

- A new attempt to consolidate Performance Engineering
  - Taking a holistic approach not limiting to technical issues only.
  - DevPerfOps manifesto states "Performance is not just a technical issue, but a business driver. We believe that performance directly impacts user satisfaction, brand reputation, and ultimately, business success."
  - https://devperfops.org

15

Shift Left
Integration with Development

16

Adjusting to Agile and CI/CD

- Agile development completely changes the performance testing dynamics
  - Working system on each iteration from the beginning
  - You need a performance engineer for the whole project
    - **Savings come from detecting problems early**
- Traditional Performance Engineering teams don't scale well
  - Increased volume exposes the problem
    - Early testing
    - Each iteration
- Remedies: automation, making performance everyone's job (->"Shift Left")

Early Testing - Mentality Change

- Making performance everyone's job
- Late record/playback performance testing -> Early Performance Engineering
- System-level requirements -> Component-level requirements
- Record/playback approach -> Programming to generate load/create stubs
- "Black Box" -> "Grey Box"

Need for Continuous Performance Testing

- Integration into Agile and CI/CD
  - To catch regressions early
  - Collecting all info needed to investigate
  - From realistic testing to coverage, from SLO checking to differences between builds
- Foundation to build further automation on the top of it
  - For further performance optimization

Continuous Performance Testing

19

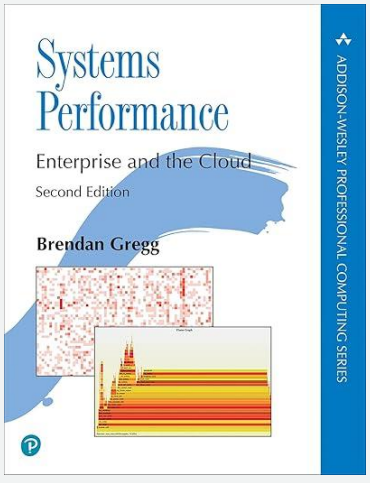
Challenges of Continuous Performance Testing

- Integration
- Coverage Optimization
- Variability / Noise Reduction
- Change Detection
- Advanced Analysis
- Operations / Maintenance

Modern Challenges in Performance Testing

20

Systems Performance Book



- 886 pages
  - Modeling, 7 pages – including USL and Queueing theory
  - Large detailed sections on OSs, CPUs, Memory, File Systems, Disks, Network, perf, Ftrace, BPF

21

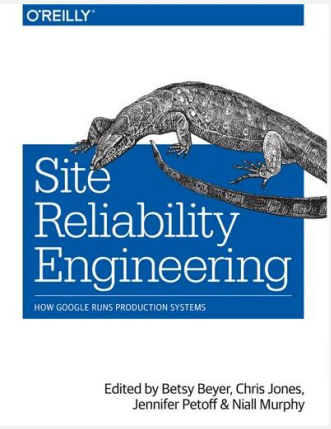
Shift Right Integration with Operations

22

Observability

- Significant advances in tools (under observability umbrella)
  - Allow to do more PE work in production
    - In spite of more sophisticated environments
  - Three pillars: metrics, logs, traces
  - Further PE integration with operations
- [Notion: Systems are too complex to properly test / reproduce problems]

Site Reliability Engineering (SRE)



- Tenets of SRE
  - "SRE team is responsible for the **availability, latency, performance, efficiency, change management, monitoring, emergency response, and capacity planning** of their service(s)."
  - None is considered separately in principles and practices
    - Except Service Level Objectives

FinOps

- FinOps Foundation www.finops.org
  - Organization, framework, certifications, etc.
- Focus on cost
- The Foundation making drastic steps to expand
  - Cost-Aware Product Decisions
  - The Scope of FinOps Extends Beyond Public Cloud

AWS Cloud Financial Management

The four pillars of Cloud Financial Management.

- From Cost Optimization Pillar
- Cloud Financial Management with AWS

Summary

- Systems scale and sophistication skyrocket – performance gets more attention
- Performance engineering is adjusting to major industry trends
  - First of all, cloud computing and agile development
- Performance engineering gets more integrated
  - Both with development (“Shift Left”) and operations (“Shift Right”)
  - More difficult to define the performance engineering body of knowledge

27

Thank you!

Alex Podelko
podealex@amazon.com

28