Pythia: A just-in-time instrumentation framework for debugging distributed systems

Lily Sturmann,

Rajul Kumar, Vladimir Pchelin, Shuwen (Jethro) Sun,
Orran Krieger, Peter Portante, Raja Sambasivan

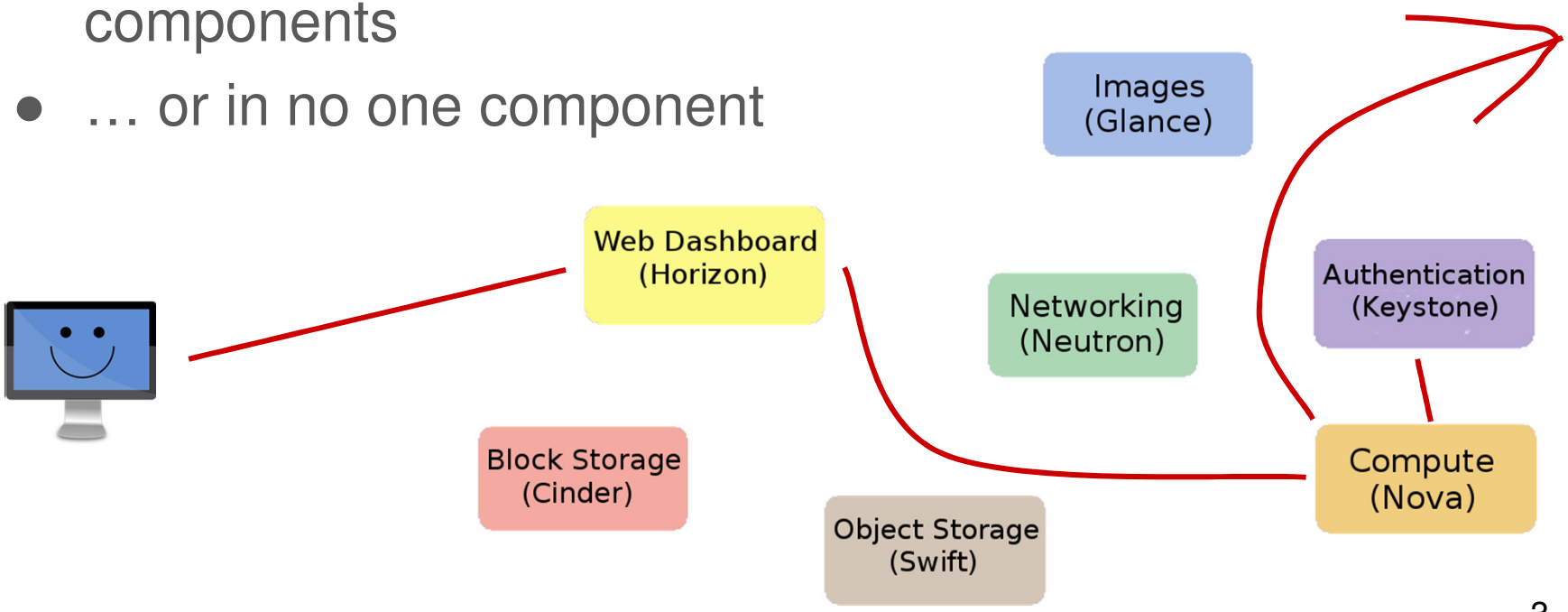
Northeastern University

Today's approach: manual, iterative instrumentation

49% of developers' time is spent debugging

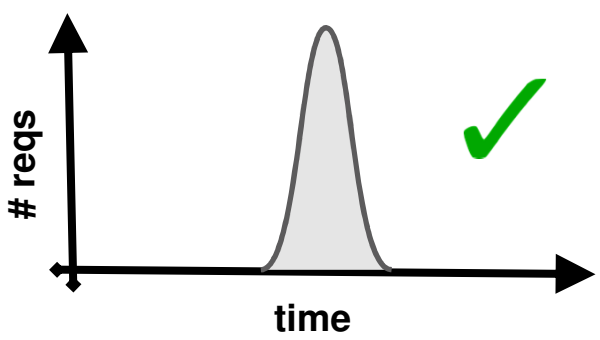
Debugging distributed systems is hard

- Problem could be in any of a large number of components
- … or in no one component

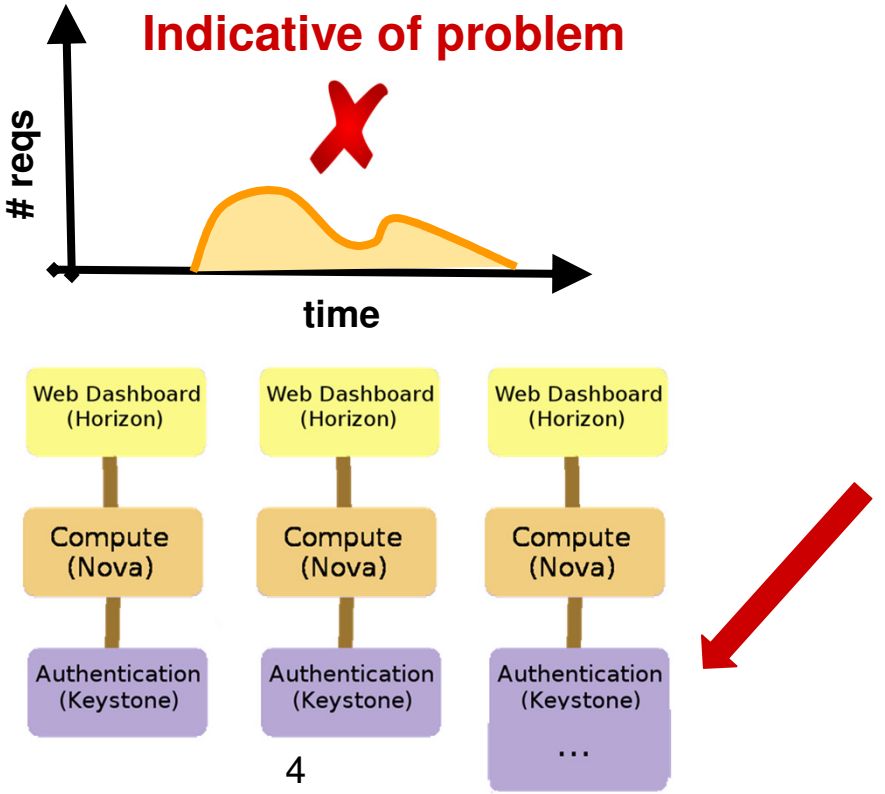


Localize variation to enable instrumentation

Similar request processing = similar performance



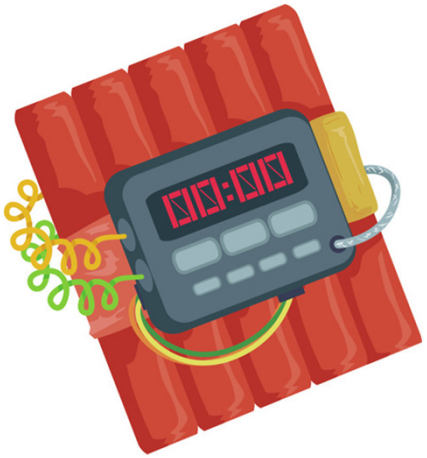
Indicative of problem



Pythia: just-in-time instrumentation framework

Our framework determines:

- Where to enable instrumentation
- What instrumentation to enable

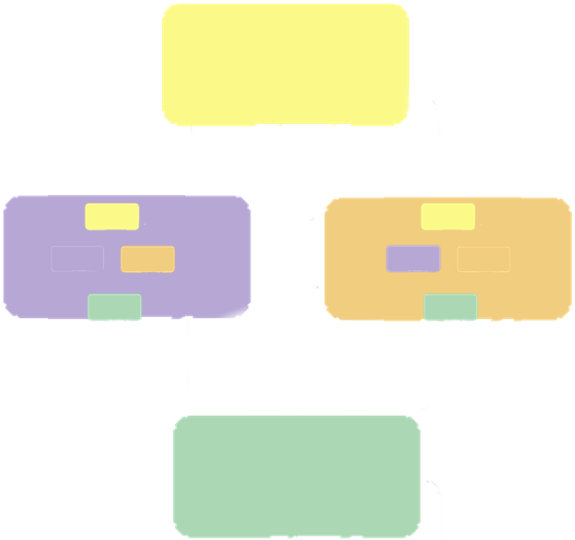


Enabling technologies:

- Ability to see request's path through system: e2e tracing
- Ability to modify or inject code into a running system

Pythia: Workflow

Obtain more detailed "skeleton" of request flow



Categorization: Group requests with same structure through system

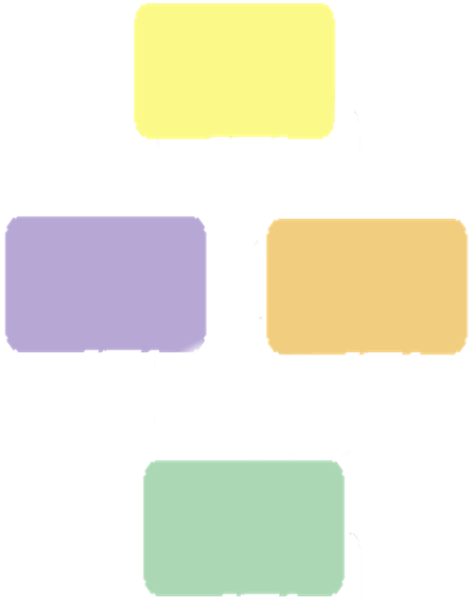
Analyze for high variance in group

Localize variation

Enable more instrumentation

Skeletons obtained via end-to-end tracing

- Captures all work done within and among components on behalf of a request
- How it works:
  - Propagates metadata along request path in the system
  - Coherently samples to limit overhead
- Used today in production (e.g., Google,



Pythia: Workflow

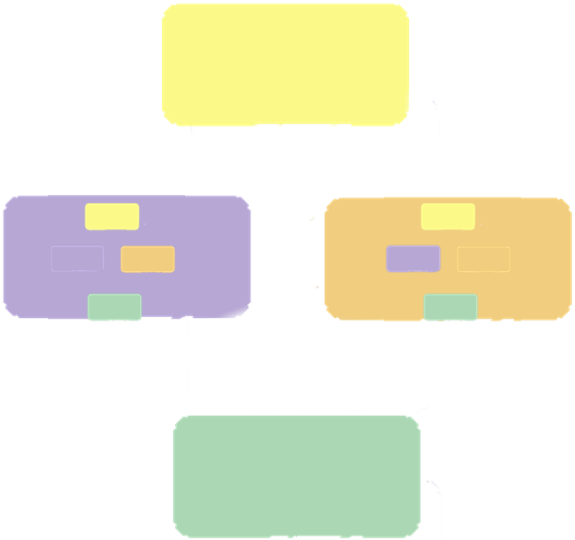
Obtain “skeleton” of request

Categorization: Group requests with same structure through system

Analyze for high variance in group

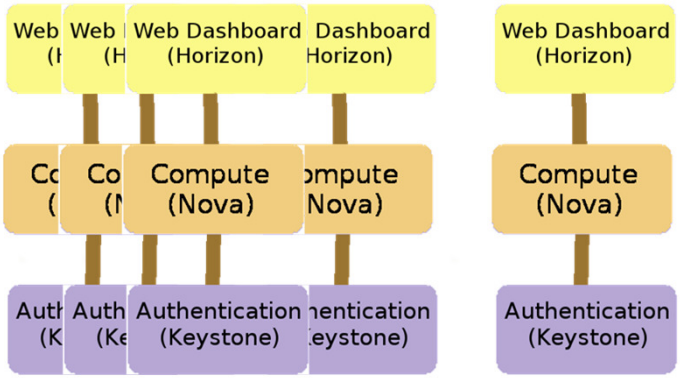
Localize variation

Enable more instrumentation



How we categorize the “skeletons” and ID high-variance categories

- Use Directed Acyclic Graph model to represent request flow
- Iteratively group request flows based on skeleton structure
- Use coefficient of variation to identify high variance within groups



Ongoing work

- Initial explorations in OpenStack
- Future explorations in microservices
- Explore enabling instrumentation across stack levels

Summary

- High variation between similar request flows may indicate performance problem or bug
- Dynamic instrumentation framework provides visibility into new problems & localizes source during runtime

- **We welcome input on interesting problems to analyze**
- **Please come see us at our ePoster!**