AGH University of Science and Technology

Evaluation Methodology of Converged Cloud Environments

Krzysztof Zieliński
Marcin Jarząb
Sławomir Zieliński
Karol Grzegorczyk
Maciej Malawski
Mariusz Zyśk

Cloud Computing

Cloud Computing enables convenient, on-demand network access to shared pool of configurable. compute, storage and network resources

- **Deployment Models –** *Private, Public and Hybrid.*
- **Service Models –** *IaaS, PaaS, SaaS.*
- **Essential characteristics** - *On-demand self service, Ubiquitous network access, Metered use, Elasticity, Resource Pooling.*
- **Common Characteristics** – *Virtualization, Service Orientation, Massive Scale, Geographic Distribution*, *Advanced Security*.

Problem statement

- Running applications over cloud environments possesses many challenges related to evaluation of performance and other non-functional requirements.
- Design common methodology for assessment and comparison of various cloud solutions.
- Methodology should contain approach for **converged** benchmarking & testing on different levels.
  i.e. IaaS and PaaS
- Consider important cloud metrics that could measure: **Performance, Scalability, Stability, and Security**
  - Addressed from provider and direct user perspective.

General overview of methodology

- Qualitative analysis - assesses features or functionalities supported by specific cloud.
- Quantitative analysis - assesses specific metrics of the system, using experimental methods, including benchmarking.
- White-box vs. Black-Box.
- Consecutive steps of cloud evaluation.

Consecutive steps of methodology

1. Definition of scope and objectives
2. Definition of aspects and metrics
3. Selection of appropriate benchmarks and tools
4. Definition of test cases and scenarios
5. Design and implementation of test scenarios and preparation of test environment
6. Execution of experiment and data acquisition
7. Analysis of results

Scope and Objectives - IaaS

- This type of service is the foundation for the whole cloud system.
- Bottlenecks in this layer would result in tremendous degradation of the responsiveness of the whole system.
- Involves testing CPU, memory, disk I/O, networking components
  - Interaction - e.g. lack of memory might reduce I/O performance of disk (due to caching issues) and network (lack of buffer space).
  - Instance size/flavor, hardware underneath, infrastructure saturation level (including instance neighbors in multitenant environment), OS configuration.

IaaS metrics

Performance	Scalability	Stability	Security	Qualitative
CPU MIPS/GOPS/FLOPS	Elastic speedup	Variability	Access protection, authentication, authorization	API support
RAM MBps	Parallel efficiency	Stability under failure	Network security	Geographical location
Disk IOPS, GB/s	Scale up-down-in-out	Isolation	Security overheads	Billing models
Network Mb/s	Throughput under response time constraints	Critical API call frequency		Virtualization type
VMs provisioned per time	Response time – throughput curve	Packet loss		Storage type
API response time	Performance per monetary unit	Delay jitter		VM import/export
Network latency				Resource Management features
VM provisioning time				Network virtualization
Image creation time				Paravirtualized interface drivers
CPU load				Memory ballooning
Volume provisioning time				Security features

Scope and Objectives - PaaS

- **Multitiered** nature of applications.
- **User scaling** is used to describe the capability of a system to scale - up/down, out/in.
- **Vertical and horizontal scaling** if the particular elements of the platform can scale.
- **Secure interactions** to understand the impact of enabling security to the overall performance.
- **Session maintenance** important factor for ensuring availability.
- **Service and data availability** large number of distributed components influence the availability of system.

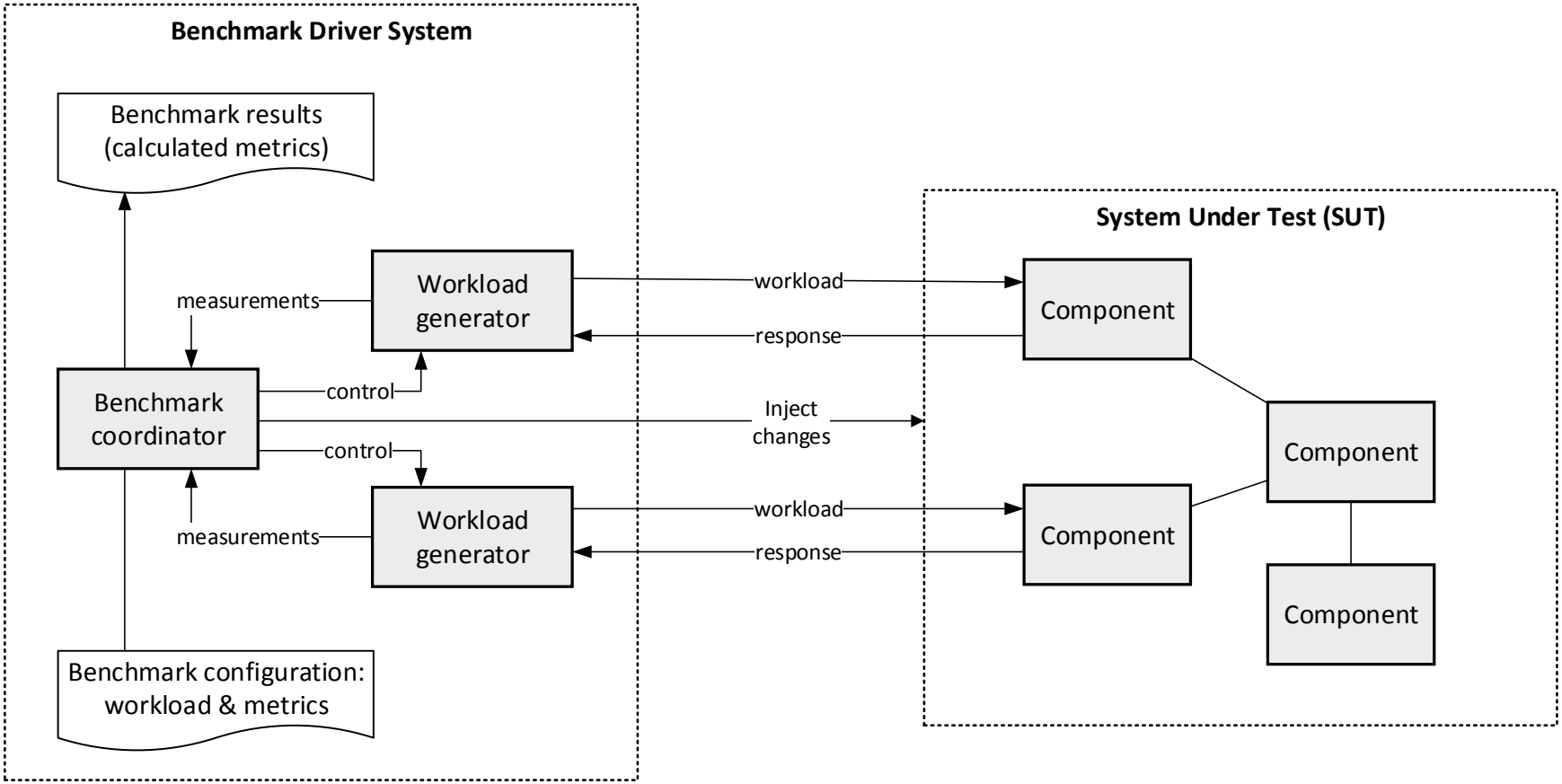
PaaS metrics

Performance	Scalability	Stability	Security	Qualitative metrics
Cloud node provisioning latency Average throughput for downloading and uploading data Application provisioning latency Application provisioning throughput Application redeploy latency	Vertical auto-scaling at cloud node level Horizontal auto-scaling at cloud node level Application scaling latency Auto-scaling at application level	Hosted application availability Service API availability Fault tolerance Maintainability	Application isolation User authorization security User data security	Platform monitoring Interface usability Supported technologies Extensibility Flexible architecture Integration Application monitoring and profiling Configuration management tools integration Application maintainability Integration with SCM systems Persistence models Geographical awareness Adjustable for hybrid clouds

Benchmarks, test cases and scenarios

- Test cases dedicated separately for IaaS and PaaS
  - IaaS – unixbench,iozone, Boniee++, ramspeed,netperf, ...
  - PaaS – CPU,Mem, Net intensive apps, Spring Travel, ...
  - Refer to different architectural components
- Evaluation of the cloud under different operational conditions
  - Stable state with constant load,
  - Very heavy load
  - Dynamically changing load

Testing environment



Summary

- Defined set of scenarios which are oriented on delivering of specific metrics
- Scenarios are parameterized, define step-by-step procedures, and summarize expected results and procedures needed for the analysis
- Allow evaluation of the system under test not only in a steady state but also provides very important information about its dynamics