How to Advance TPC Benchmarks with Dependability Aspects

Raquel Almeida[1], Meikel Poess[2], Raghunath Nambiar[3], Indira Patil[4], Marco Vieira[1]

[1] University of Coimbra, Portugal   [2] Oracle Corporation, USA

[3] Cisco Systems, Inc., USA   [4] Hewlett Packard Company, USA

rrute@dei.uc.pt, meikel.poess@oracle.com, rnambiar@cisco.com, Indira.Patil@hp.com, mvieira@dei.uc.pt

TPCTC 2010

TPCTC 2010 Singapore

Outline

- Our challenge

- The dependability benchmarking concept

- Extending the individual spec of TPC benchmarks

- Unified approach for augmenting TPC benchmarks

- Conclusion and Future steps
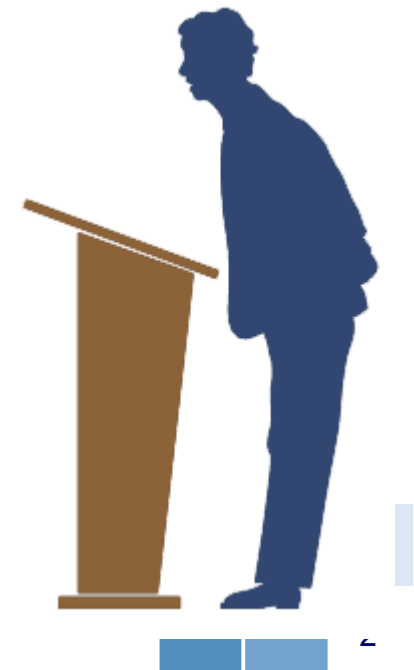
We're here to challenge TPC!!!

- Is computer benchmarking only about performance?

  - NO!!!

- e.g. nowadays most systems need to guarantee high availability and reliability

  - It is mandatory to shift the focus from measuring performance to the measurement of both performance and dependability

- Don't computers fail?

 - What is the impact of failures into the system?

The Dependability Benchmarking Concept

Procedures to measure both the dependability and performance of systems or components

- Compare systems or components from a dependability point-of-view:

  - Availability
  - Reliability
  - Safety
  - Confidentiality
  - Integrity
  - Maintainability
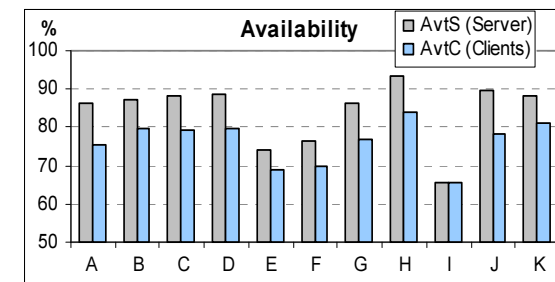
Components of a dependability benchmark
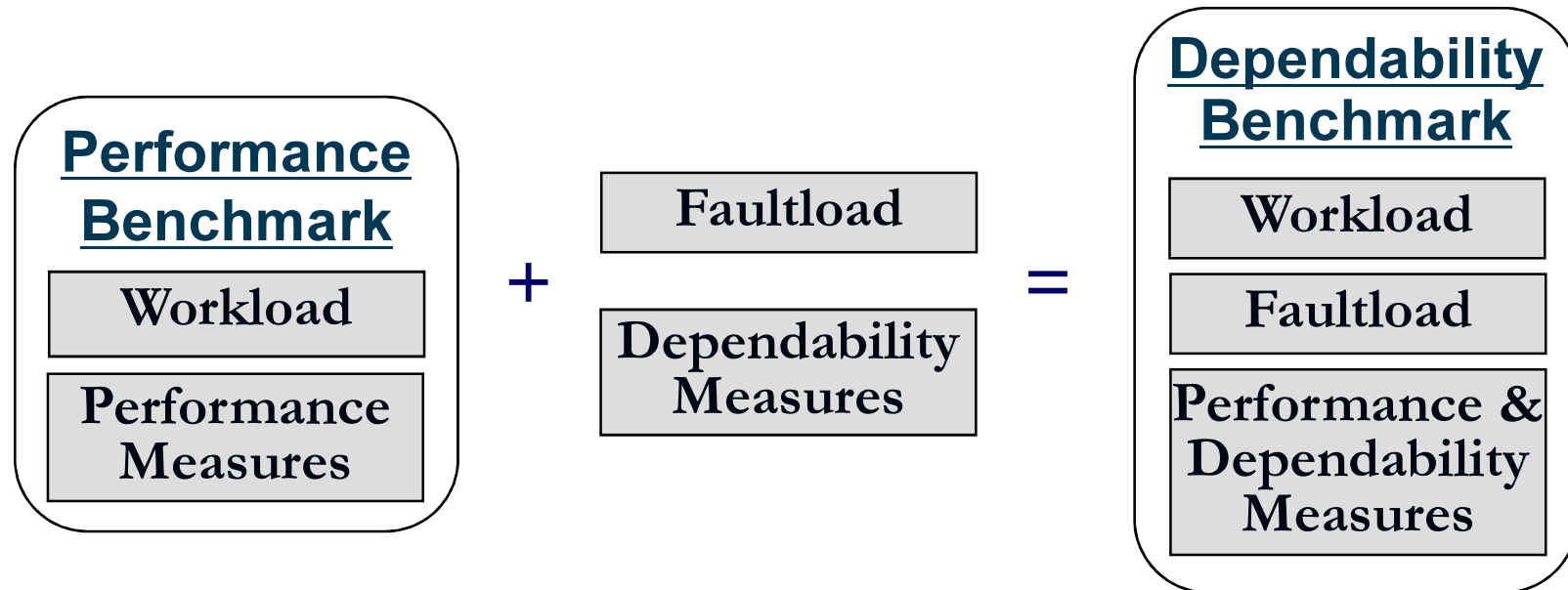
- Procedure and rules

- Experimental setup

How to Extend TPC Benchmarks?

- Take advantage of the existing ACID tests
  - Extend those tests for measuring dependability aspects

- Two approaches:

  - Extending each individual TPC specification

  - Unified approach for augmenting TPC benchmarks
    - Similar to TPC-Energy

  Let's then take a look at each approach...

Option #1: Extending each TPC Spec

- ## Two alternatives:

  - ### Modifying each specification
    - To include additional dependability related clauses

  - ### Defining an addendum to the specification
    - Specifies the additionally clauses in an independent way

- ## Pros & Cons:

  - ### Metrics and faultload can be tailored to the domain
    - Allows considering the most relevant metrics and faults

  - ### Requires repeating the definition and approval process for each benchmark
    - May be a long-term endeavor

Components…

- Setup, workload, and performance metrics from TPC specifications

- Metrics
  - Characterizing performance in the presence of faults and dependability attributes
  - Can be different for each benchmark

- Faultload
  - Based on the extension the existing ACID tests with operator faults
  - Can be different for each benchmark

Metrics

- **Baseline performance metrics**

  - The ones that already exist in the TPC benchmarks

- **Performance metrics in the presence of faults**

  - Characterize the impact of faults on the transaction execution

  - Similar to baseline performance metrics

- **Dependability related metrics**

  - Evaluate specific aspects of the system dependability

  - Many possible attributes

  - Should be a small with the most relevant ones

Faultload

- Three major types of faults:
  - Operator faults
  - Software faults
  - Hardware faults

- Studies point operator faults as the most important cause for computer system failures

- Augment the ACID tests by including situations that emulate operator mistakes
  - e.g., drop table, delete file, shutdown server

Benchmark example: DBench-OLTP

- Compare db-centric transactional systems

- Follows the style of the TPC benchmarks
  - (Customized) TPC-C workload

- Structured in Clauses:
  - Clause 1 – Preamble
  - Clause 2 – Benchmark Setup
  - Clause 3 – Benchmarking Procedure
  - Clause 4 – Measures
  - Clause 5 – Faultload
  - Clause 6 – Full Disclosure Report

Option #2: Unified Approach (1)

- Independent of the TPC benchmark
  - Tests and methodology that apply to many benchmarks
  - Successfully demonstrated by the TPC-Energy spec

- Advantages
  - "define-once-use-many-times" cost-savings
    - In terms of time in defining and implementing the specification
  - Easier for the sponsor to implement it for multiple cases
  - Specification easier to maintain and to extend for future benchmarks
  - Promotes comparison across vendors
    - Possibly even across benchmarks

Option #2: Unified Approach (2)

- ## Challenges:
  - Difficult to work within the existing constraints
  - Additions to benchmarks have to be done carefully
  - Constraints limit the scope of dependability metrics

- ## Two alternatives:
  - Dependency Level Approach
    - Dependability as a set of features that a system possesses
    - Tags the existence of dependability features
  - Dependability Metric Approach
    - Tests that "measure" dependability (secondary metric)
    - Reports existence of features and measures their performance

Dependency Level Approach

- Set of tests that must be executed to proof the existence of dependability functionalities

- Reporting metric, called "Dependency Level"

  - Number indicating how "dependent" a system is, e.g.:
    - Level 1: system is "available" through the load of the database and performance runs in the benchmark
    - Level 2: Level 1 + ACID tests demonstrated on the test DB
    - Level 3: Level 2 + Recovery times for system hardware, operating system reboot and database recovery reported during the crash durability test
    - …

  - A higher number indicates a higher level of dependability of the system

Dependability Metric Approach

- Secondary metric for all TPC benchmarks

- Tests would include the definition of the workload or faultload

- Defining each test includes the following steps:

    1. Identify dependability feature to be measured
    2. Define a test that adequately assesses the feature
    3. Define the measurement interval for the test

- Metric combines the measurements of all tests
    - e.g., simply a sum or an average, weighted or otherwise

Example: Protection Against User Errors

- **Test added to the existing ACID tests**
  - Drop a small table in the benchmark
    - "small" would need to be defined
  - Measure the time it takes to drop and restore the table
    - i.e., enable the database to use the table
  - Does not require a change to the schema or workload

- **The test can be specified by using a small table in the schema of each benchmark**
  - The choice of the table can be specified in terms of its minimum size

Conclusions

- Discussed different approaches for extending TPC benchmarks with dependability measures

- Key aspect for the future of the TPC standards

  - Industry demands metrics and methodologies for measuring dependability of transactional systems

- Two different approaches:

  - Augmenting each TPC benchmark in a customized way

  - Pursuing a unified approach

- Both approaches include the extension of existing ACID tests

Future Steps

- TPC should envisage the inclusion of dependability metrics in its benchmarks

- An incremental approach could be followed:

  - Starting from a single key metric

  - Apply the unified approach to disseminate the concept and foster the interest of vendors and purchasers

  - Extended to include more metrics

  - Augment specific TPC benchmarks to include the most relevant dependability metrics

  We will pursue this goal!

Questions & Comments

Raquel Almeida[1], Meikel Poess[2], Raghunath Nambiar[3], Indira Patil[4], Marco Vieira[1]

[1] University of Coimbra, Portugal   [2] Oracle Corporation, USA
[3] Cisco Systems Inc, USA   [4] Hewlett Packard Company, USA

rrute@dei.uc.pt, meikel.poess@oracle.com, rnambiar@cisco.com, Indira.Patil@hp.com, mvieira@dei.uc.pt