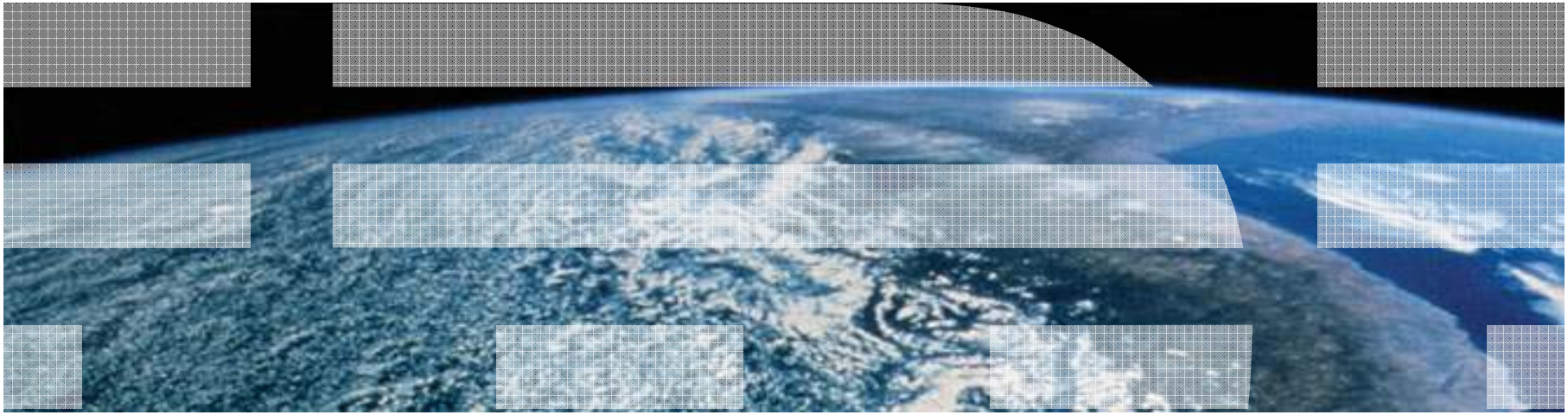
Tracing Cloud Workloads for Processor Design Evaluations

Aug 20, 2015

Saritha Vinod (IBM)
Biplob Mishra (IBM)



Agenda

- Emerging Workloads and Next Gen Processor Design
- Traces and Hardware Model
- Key Challenges in Workload Tracing
- Case-study of Cloud Data Store
- Addressing the Challenges in Trace Collection and Analysis
  - When to Take a Trace ?
  - How to Select a Trace Segment ?
  - How is Trace Generated?
  - Trace Validation and Profile Analysis
- Summary

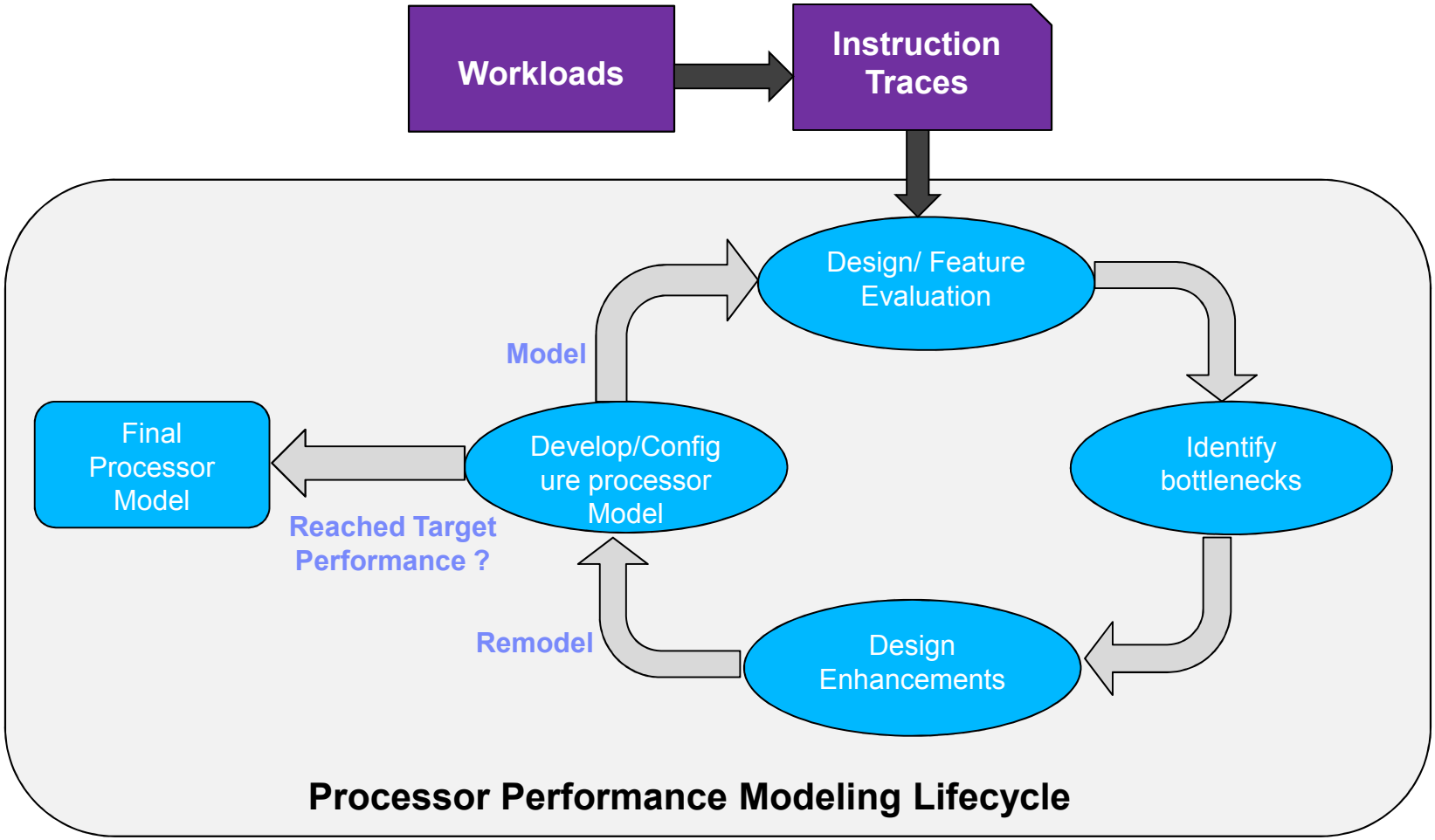
Emerging Workloads and Next Gen Processor Design

- New business trends leading to emerging workloads in domains such as big data, analytics, cloud, social and mobile

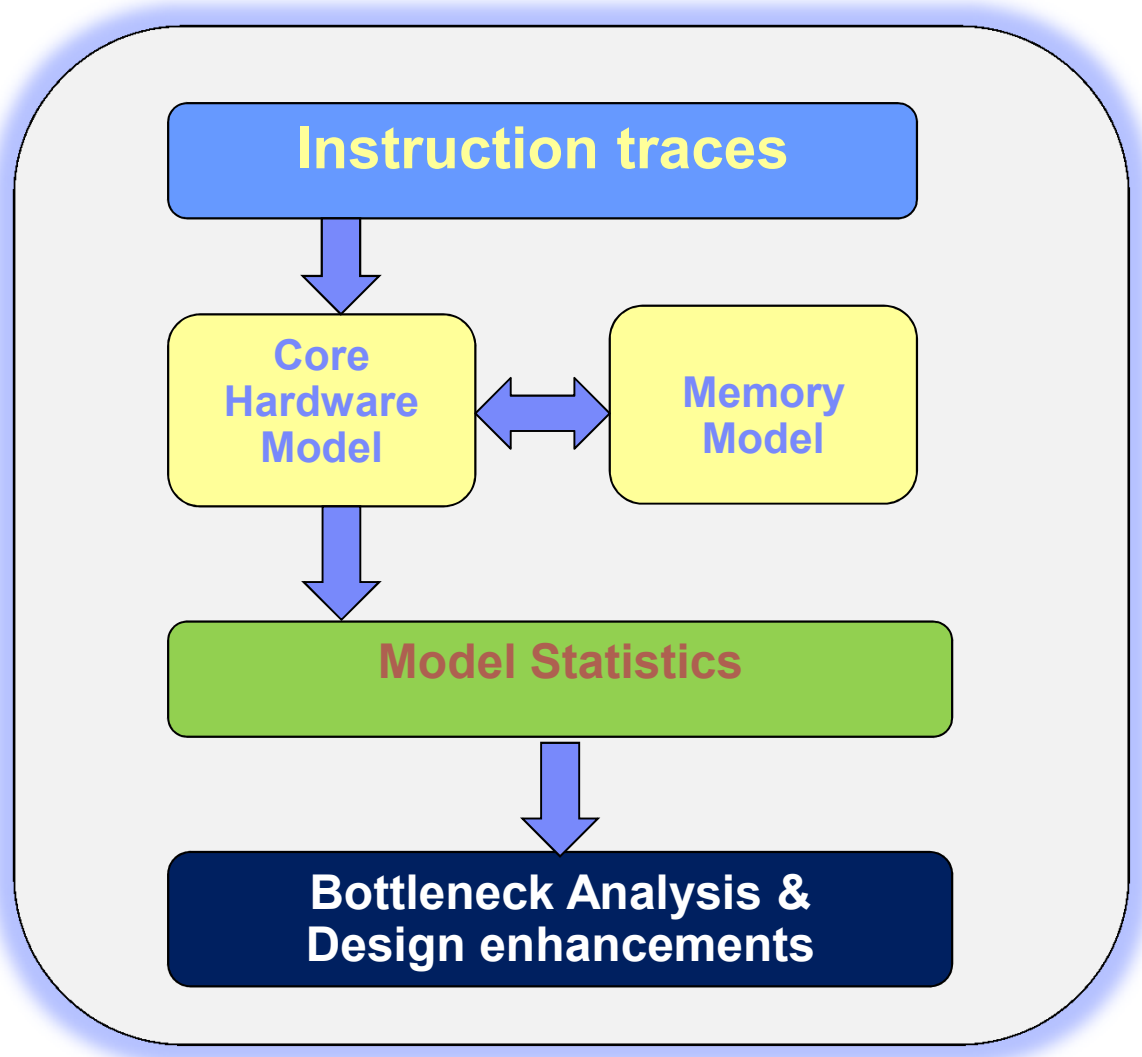
- It is important to focus on the following emerging workload characteristics while designing next generation processors
  - Instruction combinations
  - Cache access patterns
  - Data access patterns
  - Sharing of data
  - OS and Hypervisor calls
  - Instruction mixes
  - Data affinities
  - Branch related instruction mix
- How are these workload characteristics taken in as input to the processor design lifecycle ? ***Through workload traces***

Processor Performance Modeling Lifecycle

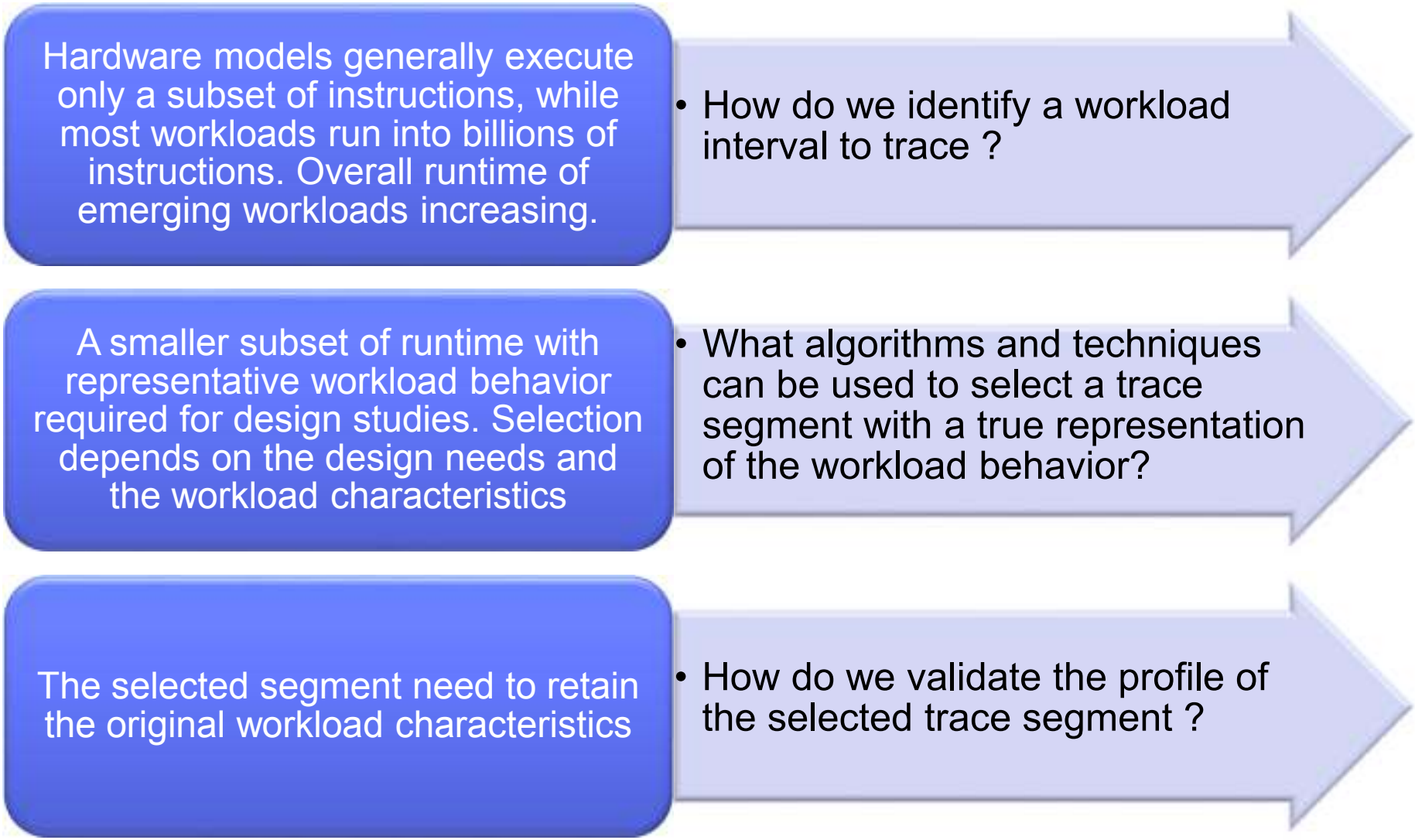
- To achieve best performance for emerging workloads the next generation processor design need to use appropriate workload traces



Traces and Hardware Model

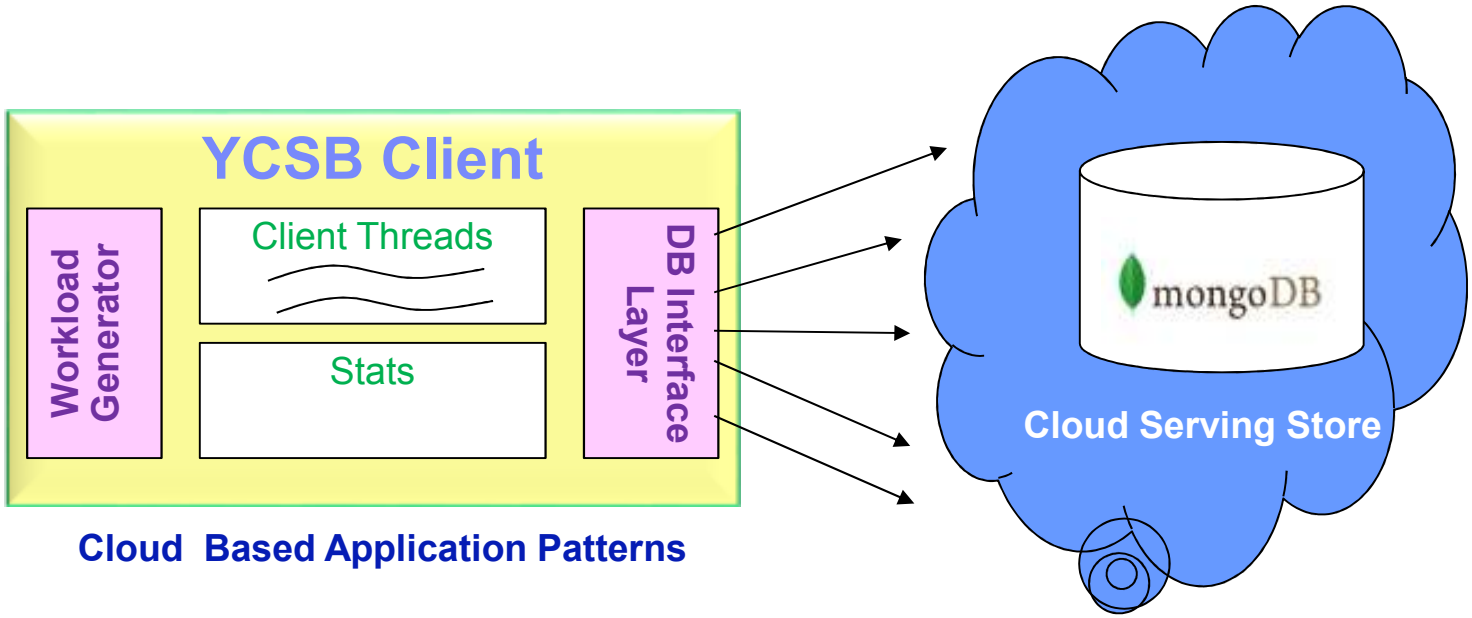


Key Challenges in Workload Tracing



Case-study of Cloud Data Store (MongoDB) Driven by a Cloud Benchmark (YCSB)

MongoDB Cloud Serving Store



Cloud Based Application Patterns

- Single MongoDB instance
  - Focus on transaction performance than scalability
- Driven through Yahoo Cloud Serving Benchmark (YCSB) Client
  - Workload generator modeling applications on cloud data system

MongoDB – Cloud Serving Database

- No-SQL, Document-oriented
- High performance, High availability, Easy scalability
- Clients connect to the MongoDB server process and perform a sequence of actions, such as inserts, queries and updates
- Stores data in files and uses memory mapped files for efficient data management
- MongoDB has multiple use cases. Analyze use cases to generate similar workload pattern

MongoDB Use Cases

- High Volume Data Feeds
- Operational Intelligence
- User Data Management
- Content Management
- Product Data

YCSB Client – Cloud Serving Benchmark Framework

Configures Cloud Workload pattern

- With a particular mix of read/write operations, data sizes and request distributions
- Used for modeling fundamental workloads generated by web applications on cloud data system

Provides Performance Benchmark Tier

- Focuses on the latency of requests when the database is under load, using constant hardware resources (scaling not considered in this tier)

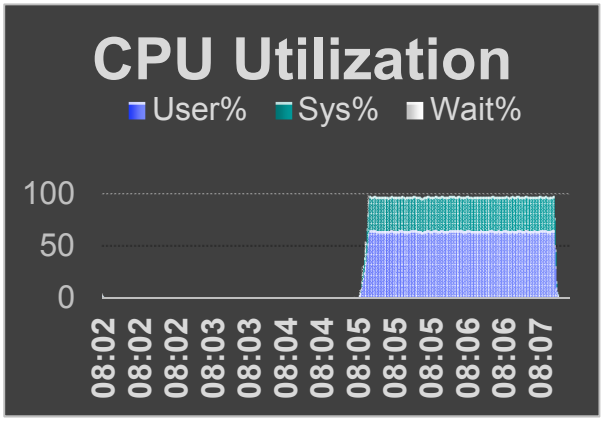
Acts as Data and Workload Generator

- Defines the dataset and load it into the database (Load Phase)
- Executes operations against the dataset while measuring performance (Transaction Phase)

Addressing the Challenges in Trace Collection and Analysis

When to Take a Trace ?

- Wait for application steady state

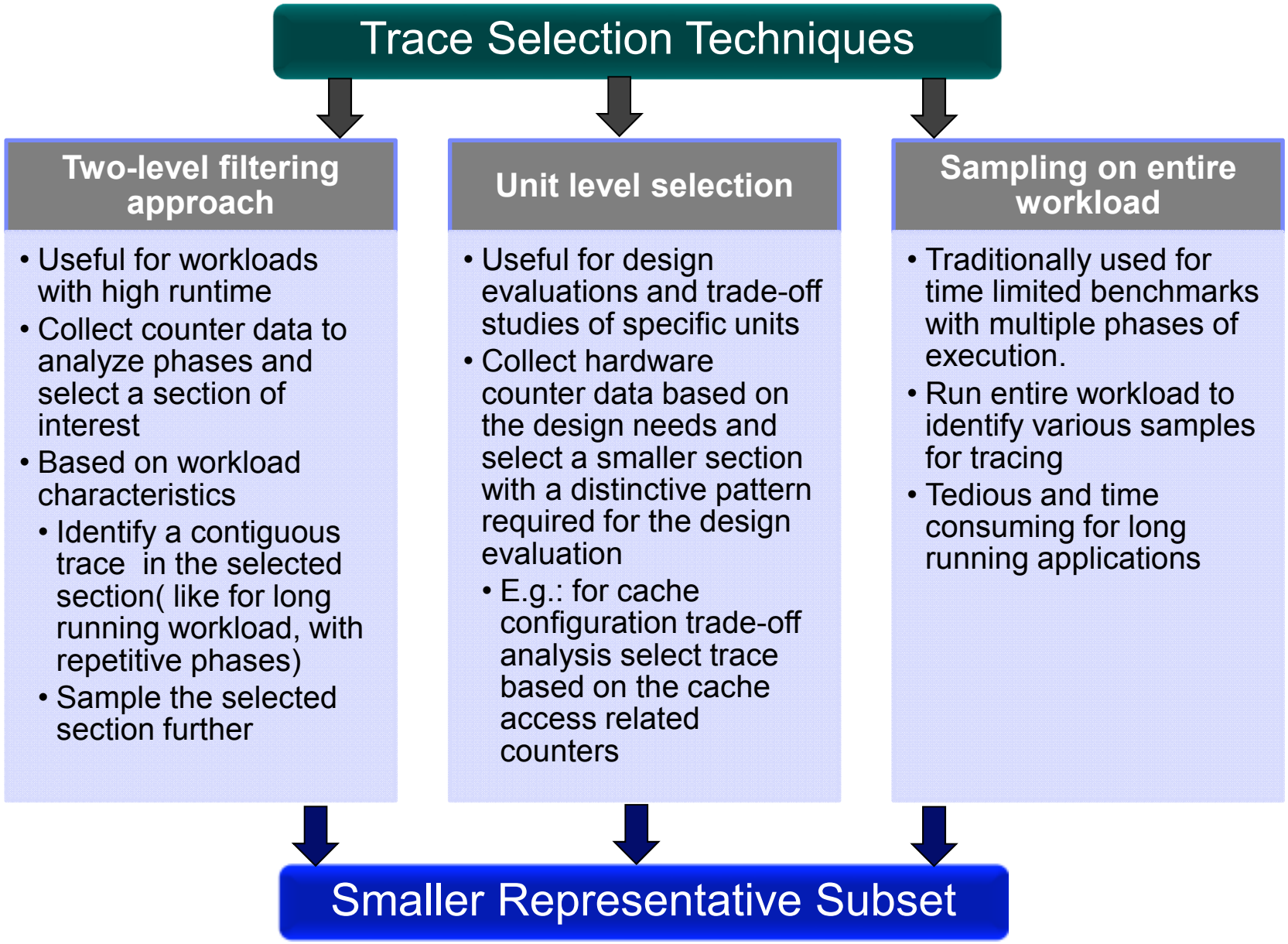


CPU utilization

Application transaction pattern. E.g.: number of records processed, throughput, latency, etc.

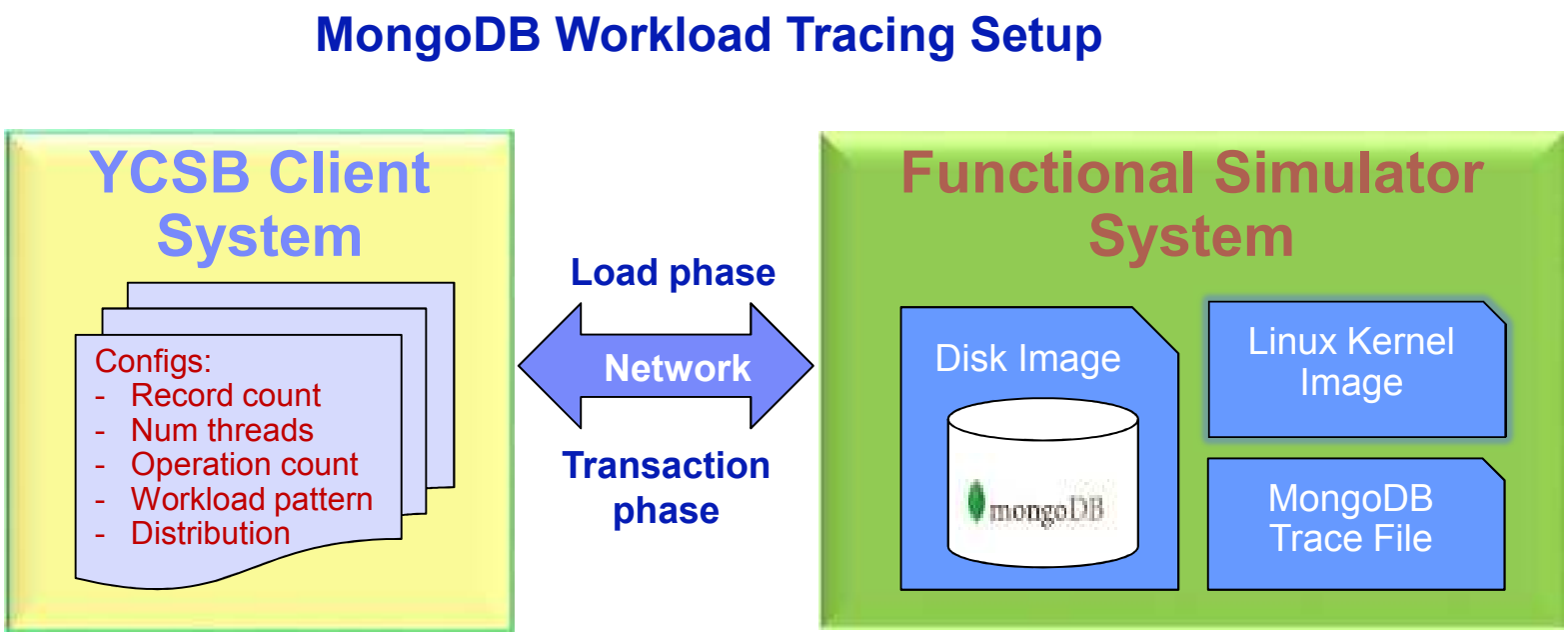
Application profile using hardware performance counters. Eg: memory bandwidth, cache misses etc.

How to Select a Trace Segment ?



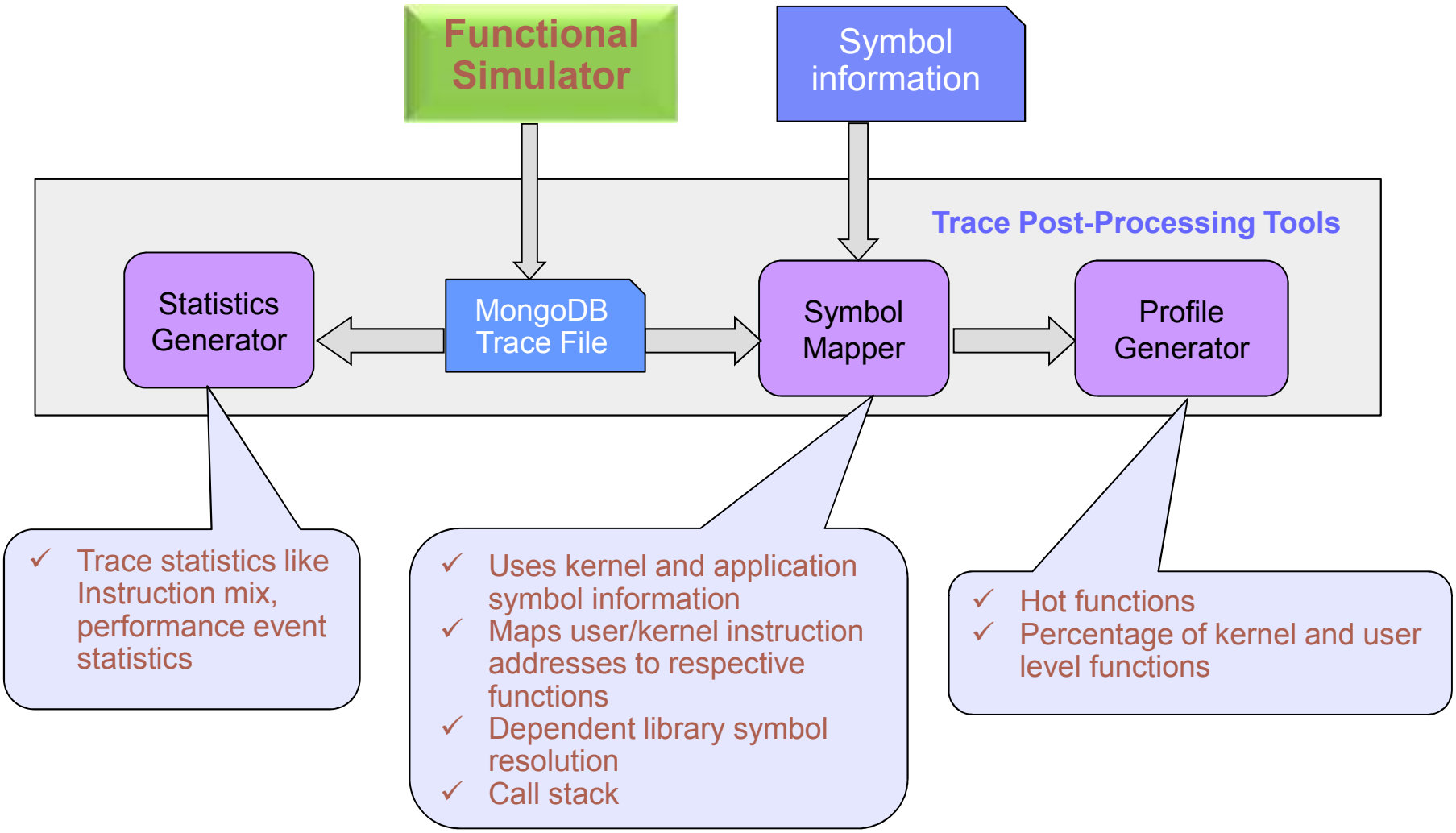
How is Trace Generated?

Setup the functional simulator to trace the selected representative segment



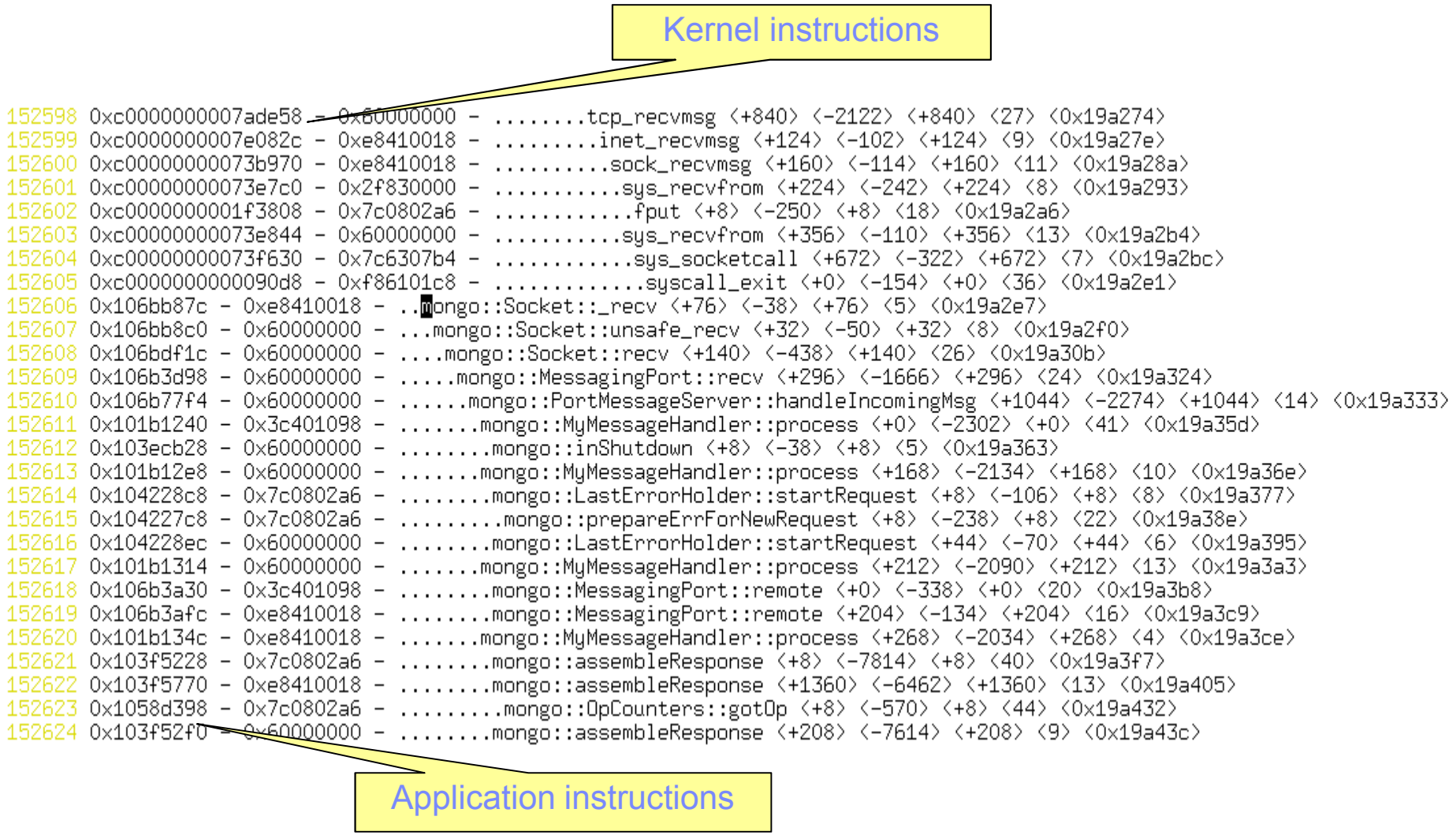
- Load phase happens first, followed by Transaction phase
- Trace only the transaction phase
- Load phase mostly repetitive in nature; Transaction phase used for performance measurement, will have interesting sequence of instructions

Trace Validation and Profile Analysis



Symbol Mapper – A sample output

- Understand the workload runtime that was captured in the trace.



Summary

- A faster drift in the newer workloads, more and more new traces need to be collected. Quality of traces and faster analysis important.
- Trace selection techniques based on a wide variety of hardware needs and workload characteristics
- Trace Validation process is gaining lot of significance; emerging workloads with complex runtime and multiple components in the application stack
- Multiple choices for trace selection:
  - Sampling on entire workload to identify phases
  - Two level filtering and contiguous trace selection(for longer runtime, transaction based workloads etc.)
  - Unit level selection (for design evaluations of specific processor features)

Questions