TPCx-BB (BIGBENCH) Big Data Analytics Benchmark

Bhaskar D Gowda | Senior Staff Engineer
Analytics & AI Solutions Group
Intel Corporation
bhaskar.gowda@intel.com
Agenda

• Big Data Analytics & Benchmarks
• Industry Standards
• TPCx-BB (BigBench)
• Profiling TPCx-BB
• Summary
The end-user Perspective of Big Data benchmarks

Analyze Volume of data in Defined SLA
Select Right Framework At Optimum TCO
What Makes a Good Benchmark?

**Comprehensive**
- Coverage (usecase, components)
- Reliable Proxy implementation
- Target usage

**Based on Open Standards**
- Peer Reviewed Spec and Code
- Public Availability
- Industry Acceptance

**Usable**
- Easy to use Kit
- Simplified Metric
- Support and Maintenance

**Flexible**
- Adaptive
- Modularized
- Reusable
Benchmark Types, use and limitation

**Micro-Benchmarks**
- Basic insights
- SPEC CPU, DFSIO
- Reduced complexity

**Functional Benchmarks**
- Specific High-level function
- Sort, Ranking
- Limited Applicability

**Application Benchmarks**
- Relevant real-world usage
- TPC-H, TPC-E
- Complex implementation
The Case for Standards and Specifications

Benchmark usage
• Performance Analysis
• Reference Architectures, Collaterals
• Influence Roadmaps and Features
• Aid Customer Deployments
• Illustrative ≠ Informative

Industry Standards
TPC
Analytics Pipeline
Big Data Benchmark Pipeline
TPCx-BB (BigBench)

Origin
- Presented in 2013 Sigmod paper¹
- Adapted from McKinsey Business cases²
- TPC standardization 2014

Features
- 30 Batch Analytic use-cases
- Framework Agnostic
- Structured, Semi-Structured & Unstructured

Collaboration
- Papers and Standardization
- Industry and Academic Presentations
- Specification and Publications

¹http://dl.acm.org/citation.cfm?id=2463712&preflayout=flat
Overview

Implementation
- Self Contained Kit
- SQL on Hadoop API’s
- Spark MLLIB Machine Learning
- Natural Language Processing

Kit Features
- Easy setup
- Size and Concurrency Scaling
- Versatile Driver
- Modular to support New API

Availability
- TPCx-BB v1
- Download from TPC
- Open to Contribution

Kit Structure

Supported Hadoop Engines

Metrics and Reporting

Modular

### Data Model

- Inspired from TPC-DS
- Semi-Structured Clickstream Data
- Unstructured Product Review Data
- Multiple Snowflake schema with shared dimensions
- Representative Skewed Database content
- Sub-linear Scaling for Non-Fact Tables
- Node Local Parallel data generation

#### TPC-DS

<table>
<thead>
<tr>
<th>Structured</th>
<th>TPCx-BB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td></td>
</tr>
<tr>
<td>Customer Address</td>
<td></td>
</tr>
<tr>
<td>Customer Demographics</td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td></td>
</tr>
<tr>
<td>Household Demographics</td>
<td></td>
</tr>
<tr>
<td>Income Band</td>
<td></td>
</tr>
<tr>
<td>Inventory</td>
<td></td>
</tr>
<tr>
<td>Item</td>
<td></td>
</tr>
<tr>
<td>Promotion</td>
<td></td>
</tr>
<tr>
<td>Web Site</td>
<td>Market Price</td>
</tr>
<tr>
<td></td>
<td>Semi-Structured</td>
</tr>
<tr>
<td></td>
<td>Click Stream</td>
</tr>
<tr>
<td></td>
<td>Un-Structured</td>
</tr>
<tr>
<td></td>
<td>Product Review</td>
</tr>
<tr>
<td>Reason</td>
<td></td>
</tr>
<tr>
<td>Ship Mode</td>
<td></td>
</tr>
<tr>
<td>Store</td>
<td></td>
</tr>
<tr>
<td>Store Returns</td>
<td></td>
</tr>
<tr>
<td>Store Sales</td>
<td></td>
</tr>
<tr>
<td>Time</td>
<td></td>
</tr>
<tr>
<td>Warehouse</td>
<td></td>
</tr>
<tr>
<td>Web Page</td>
<td></td>
</tr>
<tr>
<td>Web</td>
<td></td>
</tr>
<tr>
<td>Returns</td>
<td></td>
</tr>
<tr>
<td>Web Sales</td>
<td></td>
</tr>
</tbody>
</table>

**TPC-DS**

**TPCx-BB**
### TPCx-BB Kit Implementation

#### Example Use case
- Cluster customers into book buddies/club groups based on their in store book purchasing histories.
  - Data preparation in QL
  - Grouping in ML
- Find the categories with flat or declining sales for in store purchases during a given year for a given store.
  - Data preparation in QL
  - Regression Compute in QL

#### 1.x Kit Implementation
- Data Management expressed in SQL
- UDF & UDTF usage as necessary
- Spark ML & Deprecated Mahout
- NLP Sentiment Analysis

<table>
<thead>
<tr>
<th>Method</th>
<th>Use Case #</th>
<th>API</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query Language</td>
<td>6,7,9,11,12,13,14,15,16,17,21,22,23,24</td>
<td>SQL API</td>
</tr>
<tr>
<td>Procedural UDF/UDTF</td>
<td>1,29</td>
<td>Java UDTF</td>
</tr>
<tr>
<td>Machine Learning</td>
<td>5,20,25,26,28</td>
<td>Spark MLLIB</td>
</tr>
<tr>
<td>Procedural MR</td>
<td>30</td>
<td>Python</td>
</tr>
<tr>
<td>Procedural NLP</td>
<td>10,18,19,27</td>
<td>Open NLP</td>
</tr>
</tbody>
</table>
Benchmark Driver & Data Generator

**Benchmark Driver**
- Provide stable concurrency support
- Ensure options consistentncy
- Log runtimes
- Compute final result
- Orchestrate test
- Parse FDR Support Files
- Aggregate cluster Information

**Data Generator**
- Node Local Data Generator
- Enables independent re-generation of values
- Xml Defined and Locked
- Terabytes scale generation

---

**Java Module**
- Modular
- Invoke Benchmark
- Seeds & Metric

**Bash Scripts**
- Orchestration
- Individual Tasks
- Internode Access

**Parameters**
- Features
  - Flexible
  - Conf Files Input

---

**Data Generator**
- Schema & Format
- HDFS
Scale Factor, Streams & Metrics

**Scale Factor**
- Pre-Defined 1000-1000000
- Nominal to size in GB
- Measured at Pre-Load

**Streams**
- Parallel 1-99
- Generated with Deterministic Seed
- Randomly Placed

**Secondary Metrics**
- Computed from Phase Elapsed Times
- Elapsed Times Included in the Report
- Log File with Time Stamps

**Primary Metrics**
- Computed Performance Metric
- TPC Pricing Spec for $/Performance
- Optional Energy-Performance Metric
Test Phases

Load Test
- Data Aggregated from various sources
- Permute and Loads Aggregated Data
- Test’s Storage, Compute, Codecs, Formats

Power Test
- Each Use Case runs once
- Helps Identify Optimization areas
- Varied utilization Pattern

Throughput Test
- Multiple Jobs running in parallel
- Realistic Hadoop Cluster usage
- Optimize for Cluster Efficiency

Simulate real-world usage
Use case driven utilization pattern
Inclusive HW & SW Performance Testing
Benchmark Workflow

- **Cluster Setup**
  - Min 3 Nodes
  - Hadoop Distribution
  - Concurrent Streams 2-n

- **Define Parameters**
  - Scale Factor 1TB,3TB
  - HoMR, HoS,HoT
  - Tuning Parameters

- **Run Benchmark**
  - Data Generation
  - Load, Power, Throughput Tests

- **Analyze Data**
  - Reports
  - Metrics
  - Utilization

- **Results**
  - Publication
  - Collateral
  - POC
Profiling Using TPCx-BB

Challenges Simulation & Modeling can address

- Scale-Out distributed frameworks
- Volume, Velocity, Variety, Veracity (4v's) Data formats
- Comprehending job behaviors w.r.t. hardware & Software frameworks
- Multiple points of contention/bottleneck
- Access to suitable infrastructure
- Time consuming root causing identify and drive changes

1. Basic & Intermediate
   - Objective
     - System Sizing
     - Purchasing Decisions
     - Collaterals
   - Profiler Roles
     - Solution Architects
     - Result Publishers
     - End Users
   - Insights
     - Utilization
     - Scaling Studies
     - H/W-S/W Selection
   - Tools
     - OS utilities
     - PAT*
     - Ganglia

2. Advanced
   - Objective
     - uArch design
     - S/W Tuning
     - Projections
   - Profiler Roles
     - Platform Architects
     - S/W Developers
     - Academia
   - Insights
     - Granularity view of Platform
     - S/W stack nuances
     - Model Inputs
   - Tools
     - vTune, EMON
     - Application Logs
     - SAE/SIMICS

*https://github.com/intel-hadoop/PAT
Basic Profiling

Test Configuration
- 10 Node cluster
- Hadoop 2.X
- Hive on Spark API
- Scale Factor (SF) ~3TB
- Scaling: Stream Scaling

Test Results
- Perf. Score: BBQpm@3000 = 549
- $/BBQpm@3000 = $456
- Cluster Utilization = 70%

Data Analysis
- 9 Stream's provide best Perf. and $/Perf.
- Post 9 streams cluster hits bottleneck
- Overall Cluster Utilization ~71%
- Spark is more efficient in scaling than MR
Advanced Profiling - Platform

- Frequency Scaling of Map Reduce job
- Review with CPI, Core count etc.
- Correlate with front-end, back-end bound stages
- Memory Bandwidth studies
Advanced Profiling – Software Stack

- Data Skew issues in a Hadoop cluster
- Staggers #1 issue on large Big Data Clusters
- Larger the dataset and cluster, increased laggards tasks

Example
- 3 Tasks with data
- 1000+ Tasks launched with 0 Data
- Prolonged completion times

![Graphs showing data skew and prolonged completion times.](image-url)
TPCx-BB Challenges

• Fast Moving Big Data Landscape
• Addressing New Frameworks
• Support and Maintenance
• Large Cluster & Scale Factor Test
Summary

• Benchmark Taxonomy & Use Cases

• Industry Standard Benchmark
  ▪ Why → Common yard stick
  ▪ What → Collaborative Effort → TPCx-BB
  ▪ How → TPCx-BB 1.x Specification and Kit

• Challenges
Links

• TPCx-BB Standard specification & Kit
  http://www.tpc.org/tpcx-bb/default.asp

• Public Kit: https://github.com/intel-hadoop/Big-Data-Benchmark-for-Big-Bench

• PAT: https://github.com/intel-hadoop/PAT