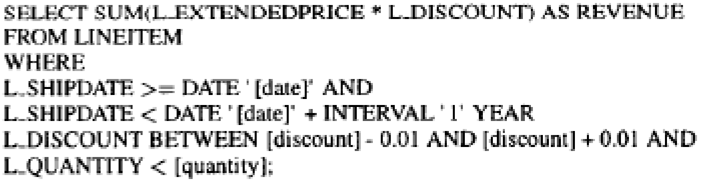
TPC Benchmarks
www.tpc.org

Current Benchmarks

- TPC-C (OLTP)
- TPC-H (DSS)
- TPC-E (OLTP)
- TPC-DS (DSS)
- TPCx-HS (Hadoop Sort)
- TPC-VMS (Run rules for virtualized runs)
- TPC – Energy (run rules for power/energy)
- TPC-BD (Big data benchmark on the way)

TPC-Queries

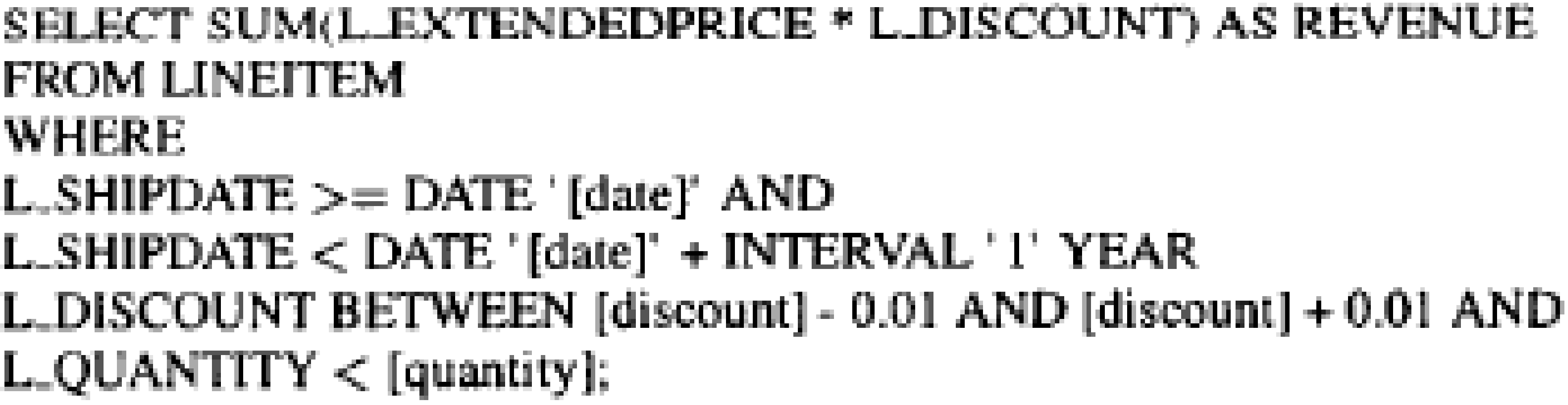
- Read and Update Queries in OLTP
- Read-only queries are more frequent in DSS
- Written in SQL (Structured Query Language)
- SQL - a non procedural language commonly used in relational databases



```
SELECT SUM(L_EXTENDEDPRICE * L_DISCOUNT) AS REVENUE
FROM LINEITEM
WHERE
L_SHIPDATE >= DATE '[date]' AND
L_SHIPDATE < DATE '[date]' + INTERVAL '1' YEAR
L_DISCOUNT BETWEEN [discount] - 0.01 AND [discount] + 0.01 AND
L_QUANTITY < [quantity];
```

TPC-Queries

- Example DSS Query



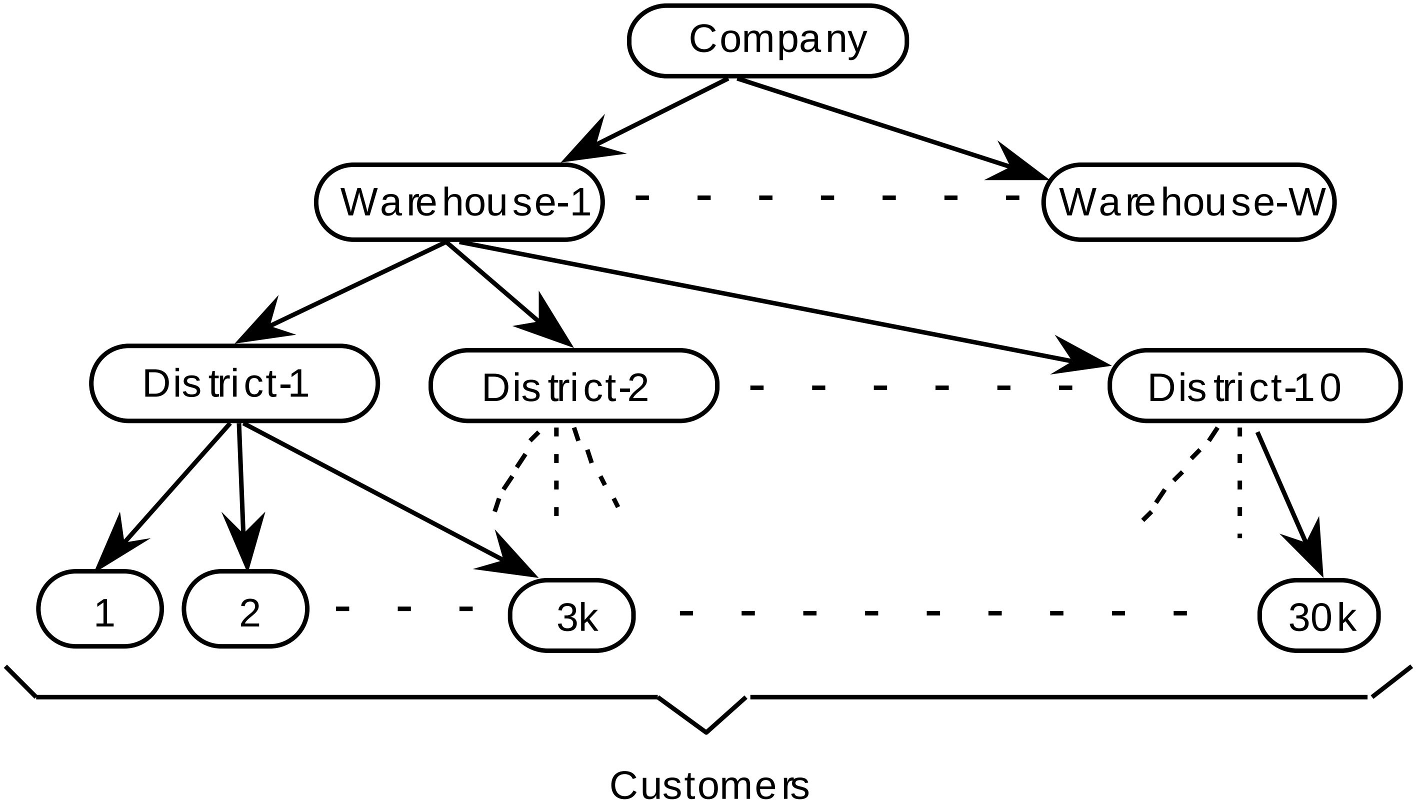
```
SELECT SUM(L_EXTENDEDPRICE * L_DISCOUNT) AS REVENUE
FROM LINEITEM
WHERE
L_SHIPDATE >= DATE '[date]' AND
L_SHIPDATE < DATE '[date]' + INTERVAL '1' YEAR
L_DISCOUNT BETWEEN [discount] - 0.01 AND [discount] + 0.01 AND
L_QUANTITY < [quantity];
```

TPC Benchmark™ C (TPC-C) – an OLTP Benchmark

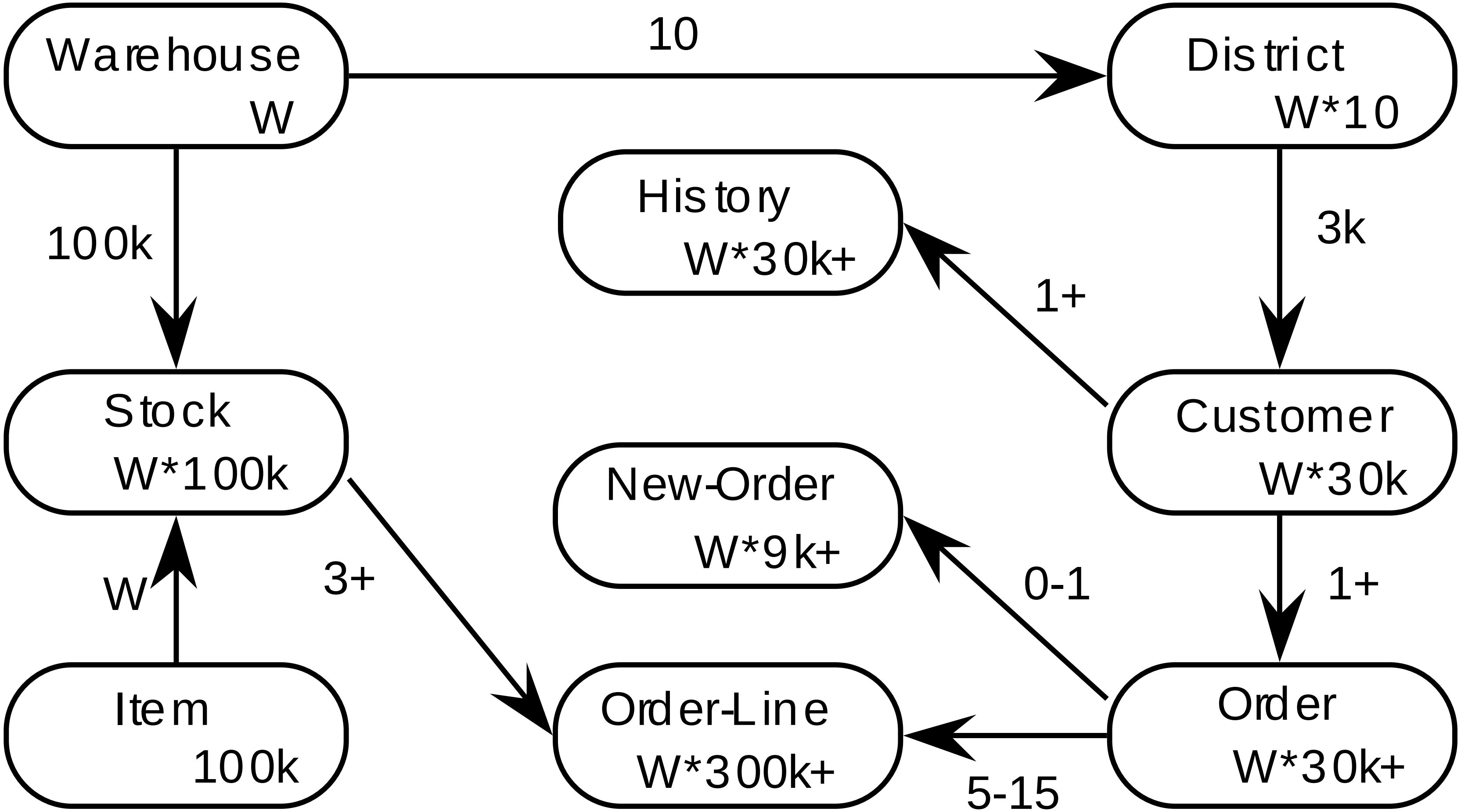
A breadth of system components

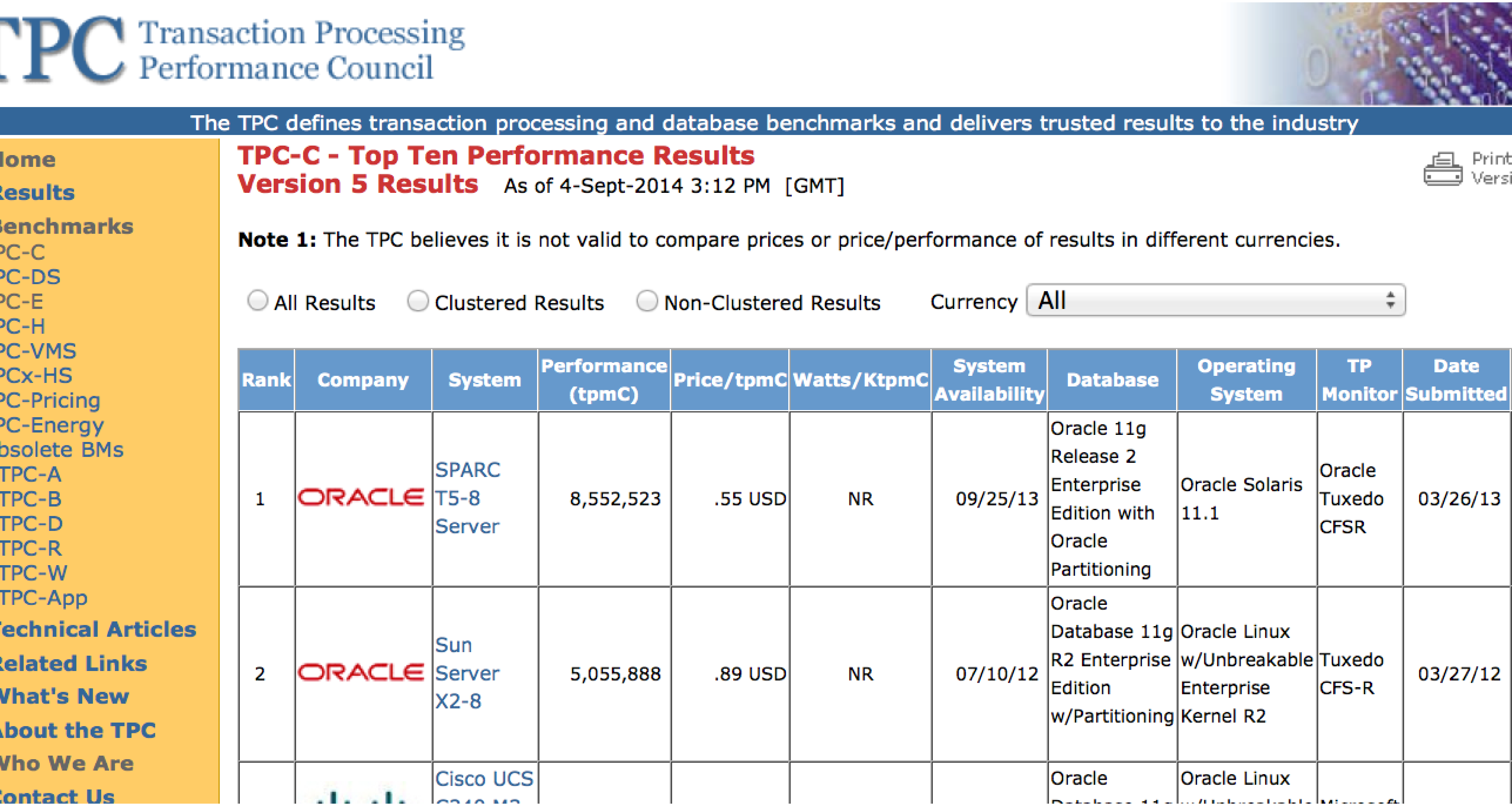
- mixture of read-only and update intensive transactions
- Contention on data access and update
- simultaneous multiple transaction types
- On-line and deferred transaction execution modes
- Multiple on-line terminal sessions
- Moderate system and application execution time
- Significant disk input/output
- Transaction integrity
- Non-uniform distribution of data access through primary and secondary keys
- Databases consisting of many tables with a wide variety of sizes, attributes, and relationships

TPC-C


Company
Warehouse-1
Warehouse-W
District-1
District-2
District-10
1
2
3k
30k
Customers

TPC-C


Warehouse
W
10
District
W*10
History
W*30k+
100k
3k
1+
Stock
W*100k
Customer
W*30k
New-Order
W*9k+
W
3+
0-1
1+
Item
100k
Order-Line
W*300k+
Order
W*30k+
5-15



TPC Transaction Processing Performance Council

The TPC defines transaction processing and database benchmarks and delivers trusted results to the industry

lome
esults
enchmarks
PC-C
PC-DS
PC-E
PC-H
PC-VMS
PCx-HS
PC-Pricing
PC-Energy
bsolete BMs
TPC-A
TPC-B
TPC-D
TPC-R
TPC-W
TPC-App
echnical Articles
elated Links
What's New
bout the TPC
Who We Are
ontact Us

TPC-C - Top Ten Performance Results

Version 5 Results As of 4-Sept-2014 3:12 PM [GMT]

Print Versi

Note 1: The TPC believes it is not valid to compare prices or price/performance of results in different currencies.

○ All Results ○ Clustered Results ○ Non-Clustered Results Currency All

Rank	Company	System	Performance (tpmC)	Price/tpmC	Watts/KtpmC	System Availability	Database	Operating System	TP Monitor	Date Submitted
1	ORACLE	SPARC T5-8 Server	8,552,523	.55 USD	NR	09/25/13	Oracle 11g Release 2 Enterprise Edition with Oracle Partitioning	Oracle Solaris 11.1	Oracle Tuxedo CFSR	03/26/13
2	ORACLE	Sun Server X2-8	5,055,888	.89 USD	NR	07/10/12	Oracle Database 11g R2 Enterprise Edition w/Partitioning	Oracle Linux w/Unbreakable Enterprise Kernel R2	Tuxedo CFS-R	03/27/12
		Cisco UCS C240 M3					Oracle Database 11g	Oracle Linux w/Unbreakable	Microsoft	

TPC Benchmark™ C (TPC-C) Metrics

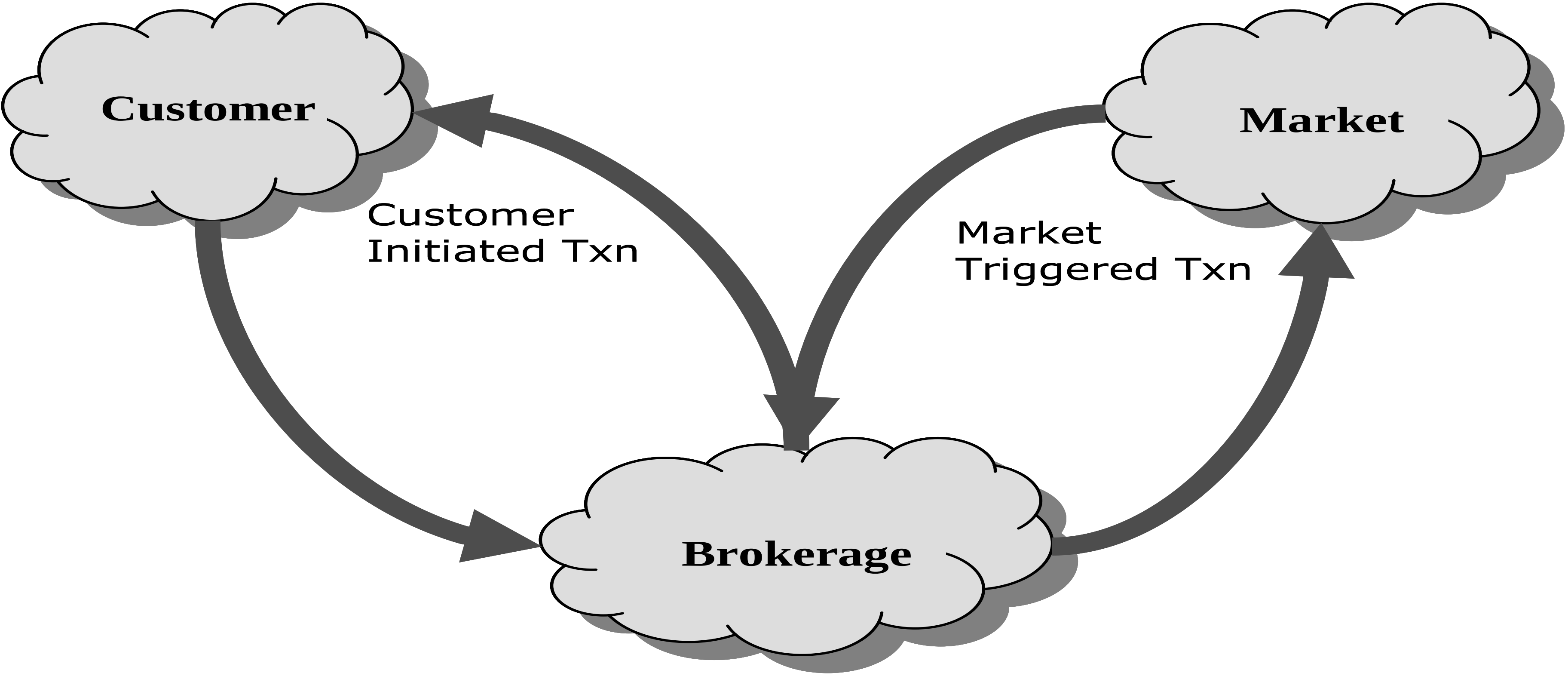
Metric: Business Throughput

the number of orders processed per minute

transactions-per-minute-C (tpmC)

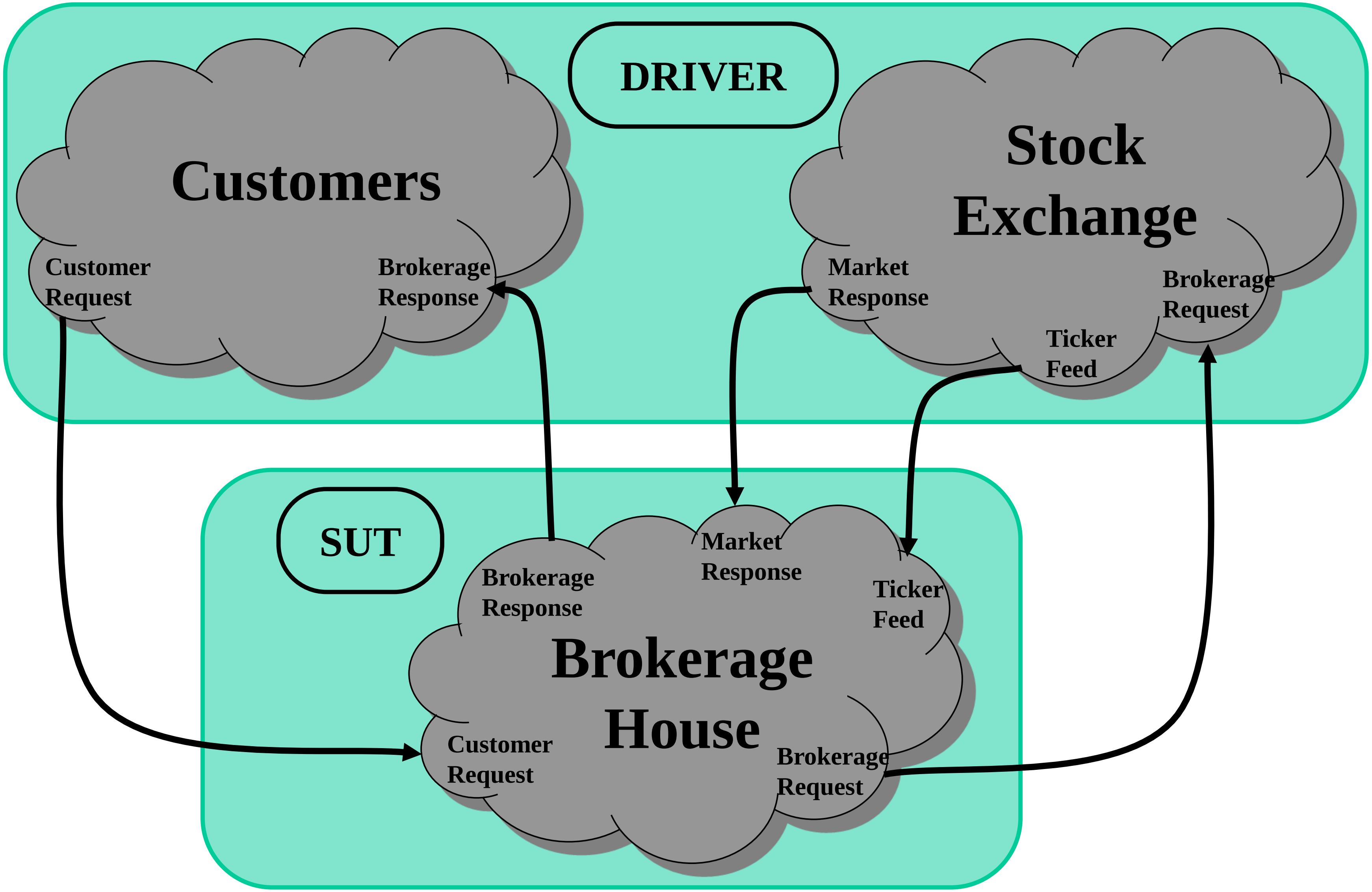
additional primary metric, expressed as watts-per-tpmC

TPC-E

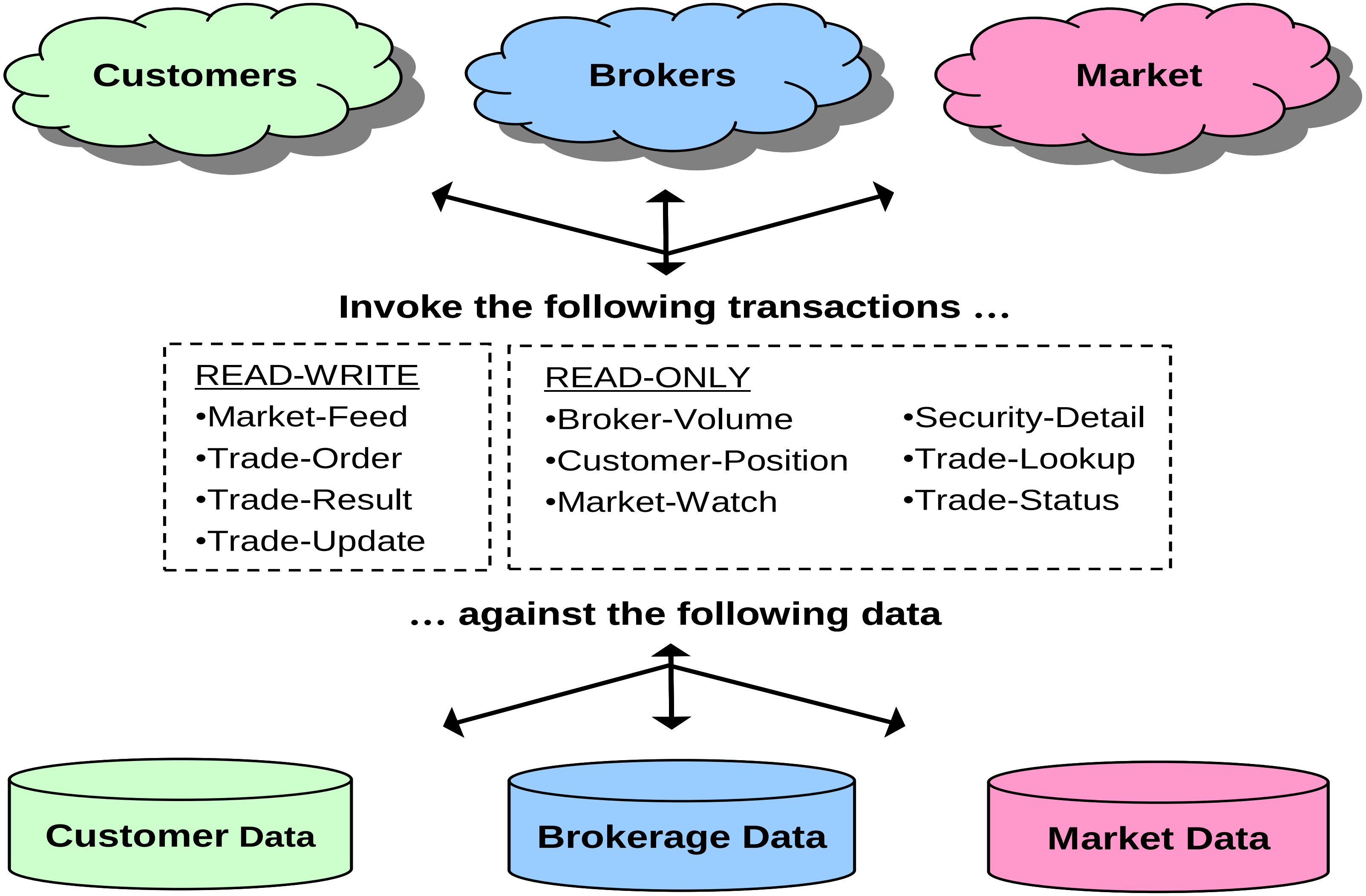


The primary metrics for TPC-E are tpsE, $/tpsE and availability date. TPC-E measures transactions per second (tpsE).

Business Model – Financial Market



TPC-E



Customers

Brokers

Market

Invoke the following transactions …

READ-WRITE

- Market-Feed
- Trade-Order
- Trade-Result
- Trade-Update

READ-ONLY

- Broker-Volume
- Customer-Position
- Market-Watch
- Security-Detail
- Trade-Lookup
- Trade-Status

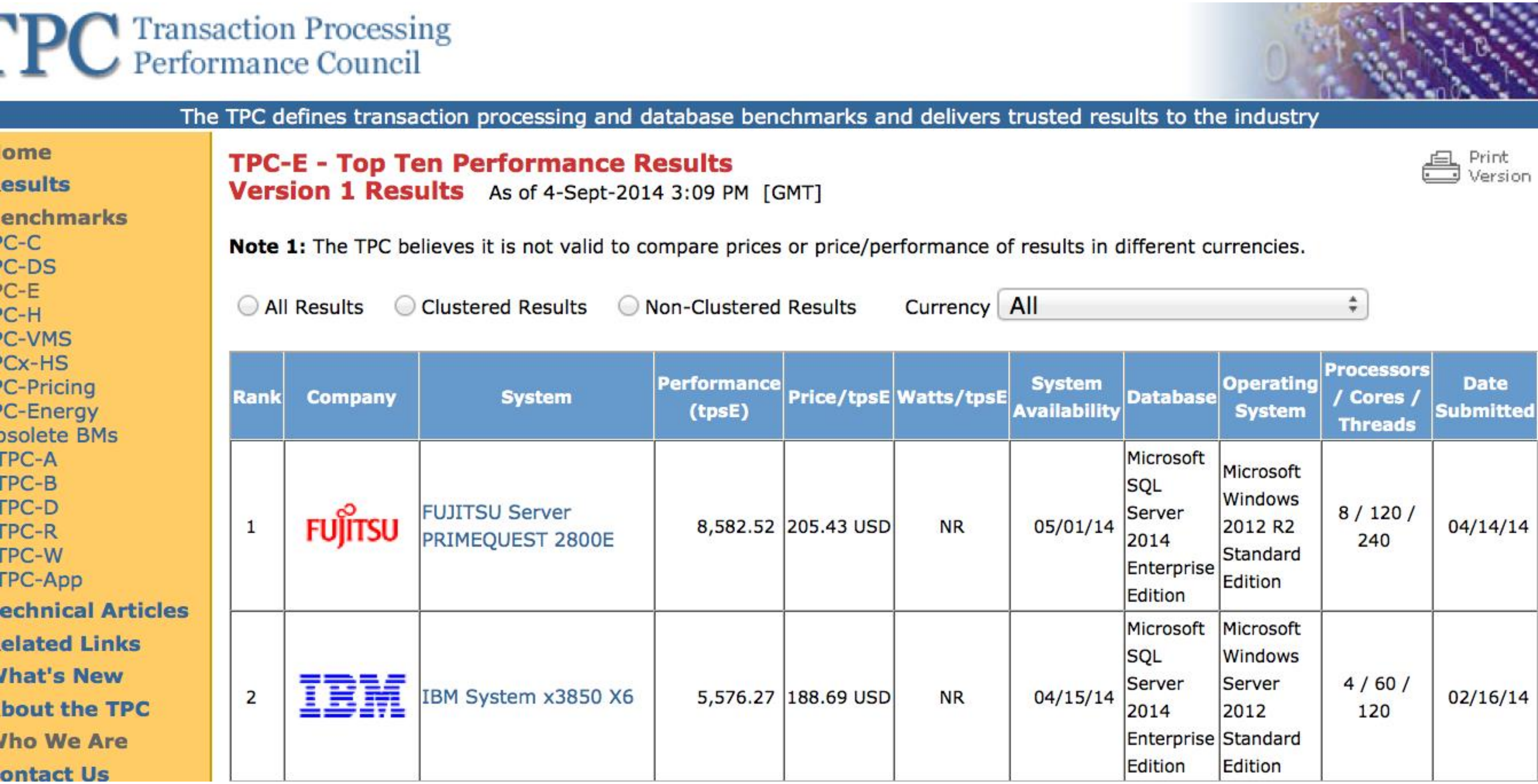
… against the following data

Customer Data

Brokerage Data

Market Data

TPC-E has 33 tables.



The TPC defines transaction processing and database benchmarks and delivers trusted results to the industry

TPC-E - Top Ten Performance Results
Version 1 Results
As of 4-Sept-2014 3:09 PM [GMT]

Note 1: The TPC believes it is not valid to compare prices or price/performance of results in different currencies.

○ All Results ○ Clustered Results ○ Non-Clustered Results Currency All

Rank	Company	System	Performance (tpsE)	Price/tpsE	Watts/tpsE	System Availability	Database	Operating System	Processors / Cores / Threads	Date Submitted
1	FUJITSU	FUJITSU Server PRIMEQUEST 2800E	8,582.52	205.43 USD	NR	05/01/14	Microsoft SQL Server 2014 Enterprise Edition	Microsoft Windows 2012 R2 Standard Edition	8 / 120 / 240	04/14/14
2	IBM	IBM System x3850 X6	5,576.27	188.69 USD	NR	04/15/14	Microsoft SQL Server 2014 Enterprise Edition	Microsoft Windows Server 2012 Standard Edition	4 / 60 / 120	02/16/14

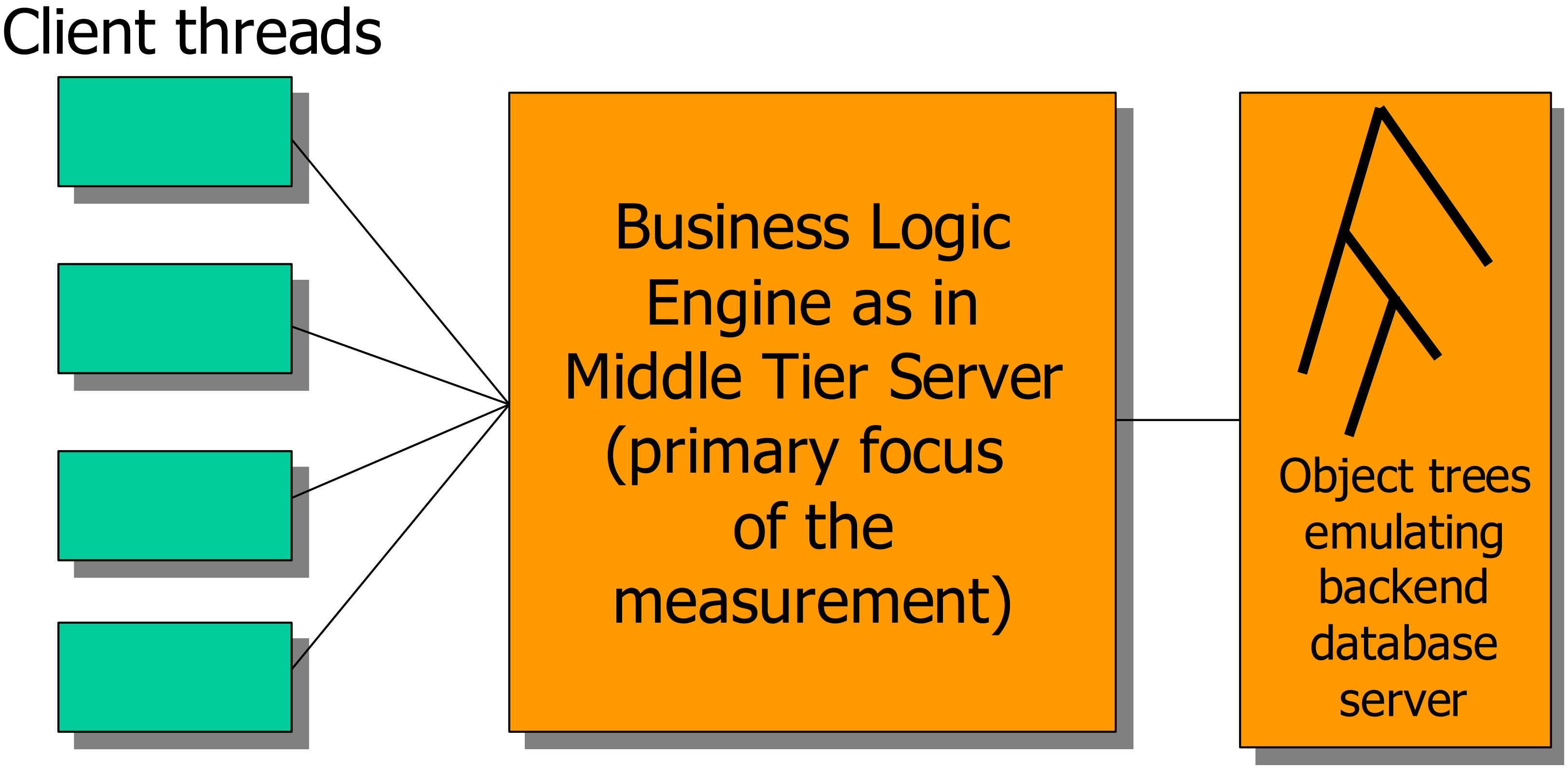
Database Tables

Customer	Broker	Market	
ACCOUNT_PERMISSION	BROKER	COMPANY	SECTOR
CUSTOMER	CASH_TRANSACTION	COMPANY_COMPETITOR	SECURITY
CUSTOMER_ACCOUNT	CHARGE	DAILY_MARKET	
CUSTOMER_TAXRATE	COMMISSION_RATE	EXCHANGE	
HOLDING	SETTLEMENT	FINANCIAL	Dimension
HOLDING_HISTORY	TRADE	INDUSTRY	ADDRESS
HOLDING_SUMMARY	TRADE_HISTORY	LAST_TRADE	STATUS_TYPE
WATCH_ITEM	TRADE_REQUEST	NEWS_ITEM	TAXRATE
WATCH_LIST	TRADE_TYPE	NEWS_XREF	ZIP_CODE

Database Scaling

Customer	Broker	Market	
🟩 ACCOUNT_PERMISSION	🟩 BROKER	🟩 COMPANY	🟨 SECTOR
🟩 CUSTOMER	🟪 CASH_TRANSACTION	🟩 COMPANY_COMPETITOR	🟩 SECURITY
🟩 CUSTOMER_ACCOUNT	🟨 CHARGE	🟩 DAILY_MARKET	
🟩 CUSTOMER_TAXRATE	🟨 COMMISSION_RATE	🟨 EXCHANGE	
🟩 HOLDING	🟪 SETTLEMENT	🟩 FINANCIAL	Dimension
🟩 HOLDING_HISTORY	🟪 TRADE	🟨 INDUSTRY	🟩 ADDRESS
🟩 HOLDING_SUMMARY	🟪 TRADE_HISTORY	🟩 LAST_TRADE	🟨 STATUS_TYPE
🟩 WATCH_ITEM	🟪 TRADE_REQUEST	🟩 NEWS_ITEM	🟨 TAXRATE
🟩 WATCH_LIST	🟨 TRADE_TYPE	🟩 NEWS_XREF	🟨 ZIP_CODE
Legend:	🟨 Fixed Tables	🟪 Growing Tables	🟩 Scaling Tables

SPECjbb

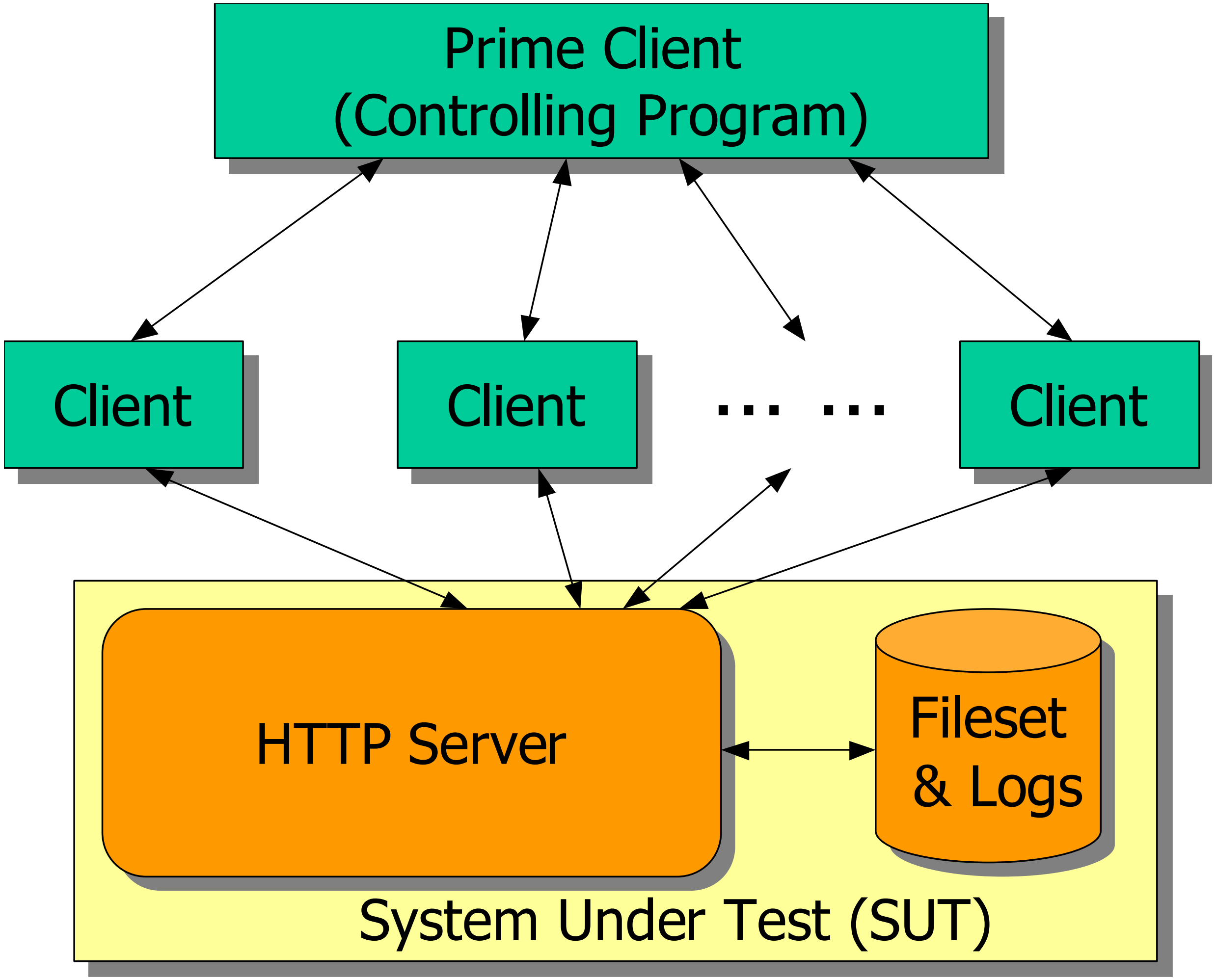


Client threads

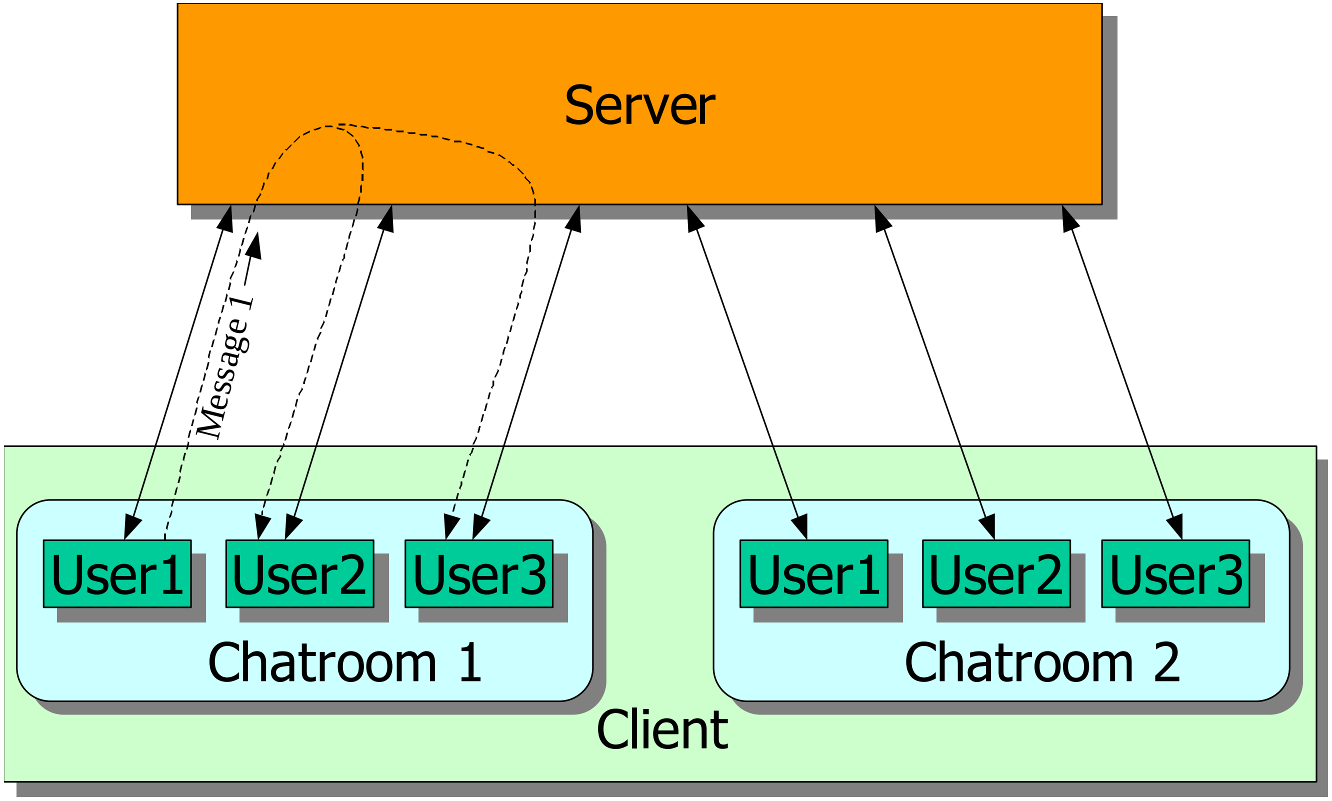
Business Logic Engine as in Middle Tier Server (primary focus of the measurement)

Object trees emulating backend database server

SPECweb


Prime Client
(Controlling Program)
Client
Client
… …
Client
HTTP Server
Fileset
& Logs
System Under Test (SUT)

VolanoMark


Server
Message 1
User1
User2
User3
Chatroom 1
User1
User2
User3
Chatroom 2
Client

Embedded Benchmarks

- EEMBC Benchmarks
- BDTI Benchmarks
- MediaBench
- MiBench
- MorphMark
- MIDP Mark
- Android Benchmarks

Embedded Benchmarks

- EEMBC Benchmarks
- BDTI Benchmarks
- MediaBench
- MiBench
- MorphMark
- MIDP Mark

EEMBC BENCHMARKS

SYSTEM

Android, Smartphone Browsers and Java

AndEBench™ - Provides a standardized, industry-accepted method of evaluating Android platform performance.

BrowsingBench® (Version 2.0 in development) - Evaluate Web browser performance on Smartphones and other systems.

GrinderBench™ - Test a system's performance running Java applications.

Telecom/Networking System Benchmarks

ETCPBench™ - TCP/IP performance and conformance for platforms from micro-controllers to high-end processors.

DPIBench™ (in development) - Evaluate throughput and latency to highlight the strengths and weaknesses of DPI systems, processors, and middleware.

Benchmark Testing Services

EEMBC Technology Center (ETC) - Services include porting and benchmark execution, performance analysis, and preparation of platforms for benchmark score certification. These services allow any company to make EEMBC benchmark testing an integral part of its product development and release process - without tying up internal engineering resources.

PROCESSOR

Microprocessor Benchmark Suites

AutoBench™ (Version 2.0 in development) - Automotive, industrial, and general-purpose applications.

ConsumerBench™ - Digital cameras, printers, and other embedded systems doing digital imaging tasks.

CoreMark® - Single number score for quick comparison of processor and microcontroller core functionality.

DENBench™ - Digital entertainment products such as smartphones, MP3 players, digital cameras, TV set-top boxes, and in-car entertainment systems.

EnergyBench™ - Power and energy performance with insights to power budget costs.

FPBench™ (in development) - Floating-point performance in graphics, audio, motor control, and other high-end processing tasks.

MultiBench™ - Multicore architectures, memory bottlenecks, OS thread scheduling, and efficiency of synchronization.

Networking - Moving and analyzing packets in networking applications.

OABench™ - Office Automation tasks in printers, plotters, and other systems that handle text and image processing tasks.

TeleBench™ - Telecommunications processors in modem, xDSL, and related fixed-telecom applications.

BDTI Benchmarks

- FIR Filter
- IIR Filter
- FFT
- Dot Product
- Viterbi Decoder

…etc

MediaBench Benchmarks

- JPEG
- MPEG
- GSM
- G.721 Voice Compression
- Ghostscript
- ADPCM (Adaptive Differential Pulse Code Modulation)

MiBench Benchmarks

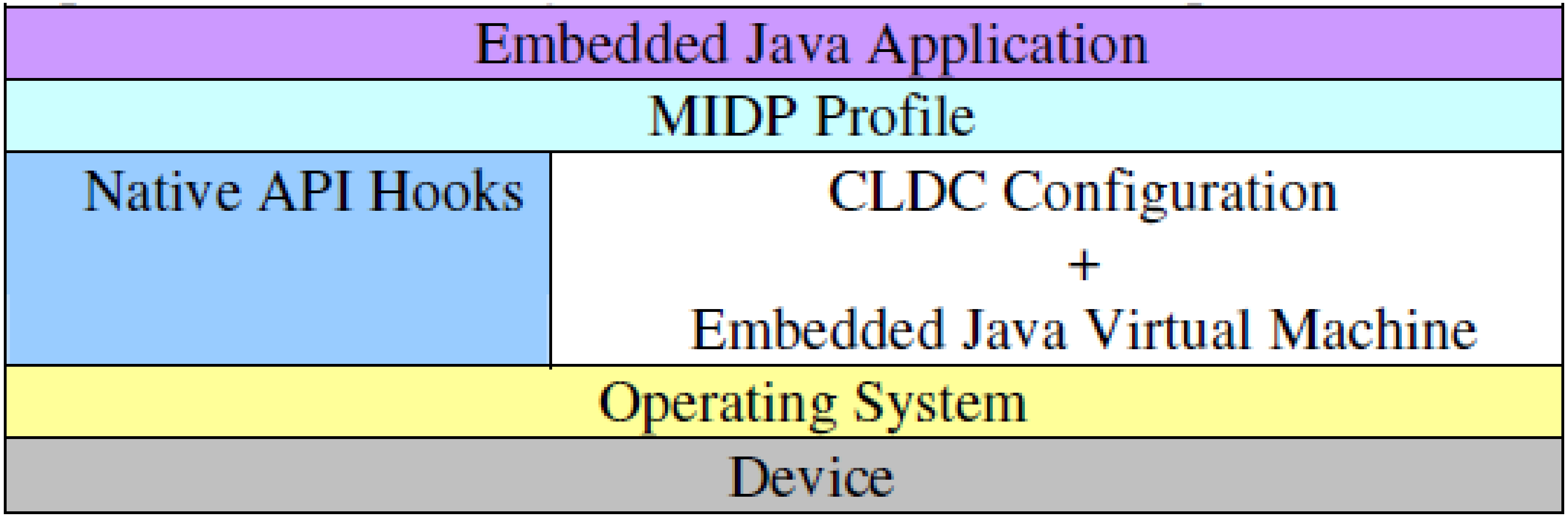
- 35 embedded programs similar to the EEMBC Programs
- First Presented in ACES Avaya at WWC 2001 (precursor to IISWC)
- Michigan – Prof. Todd Austin’s Group
- C Source
- Automotive
- Consumer
- Network
- Office
- Security
- Telecom

Morphmark

- Targets Java enabled mobile handsets
- Around 2002
- Tests
  - JVM on the handset
  - Graphics on the handset
  - Java I/O

Embedded Java

- Software Layers in Embedded Java Environment



Embedded Java Application

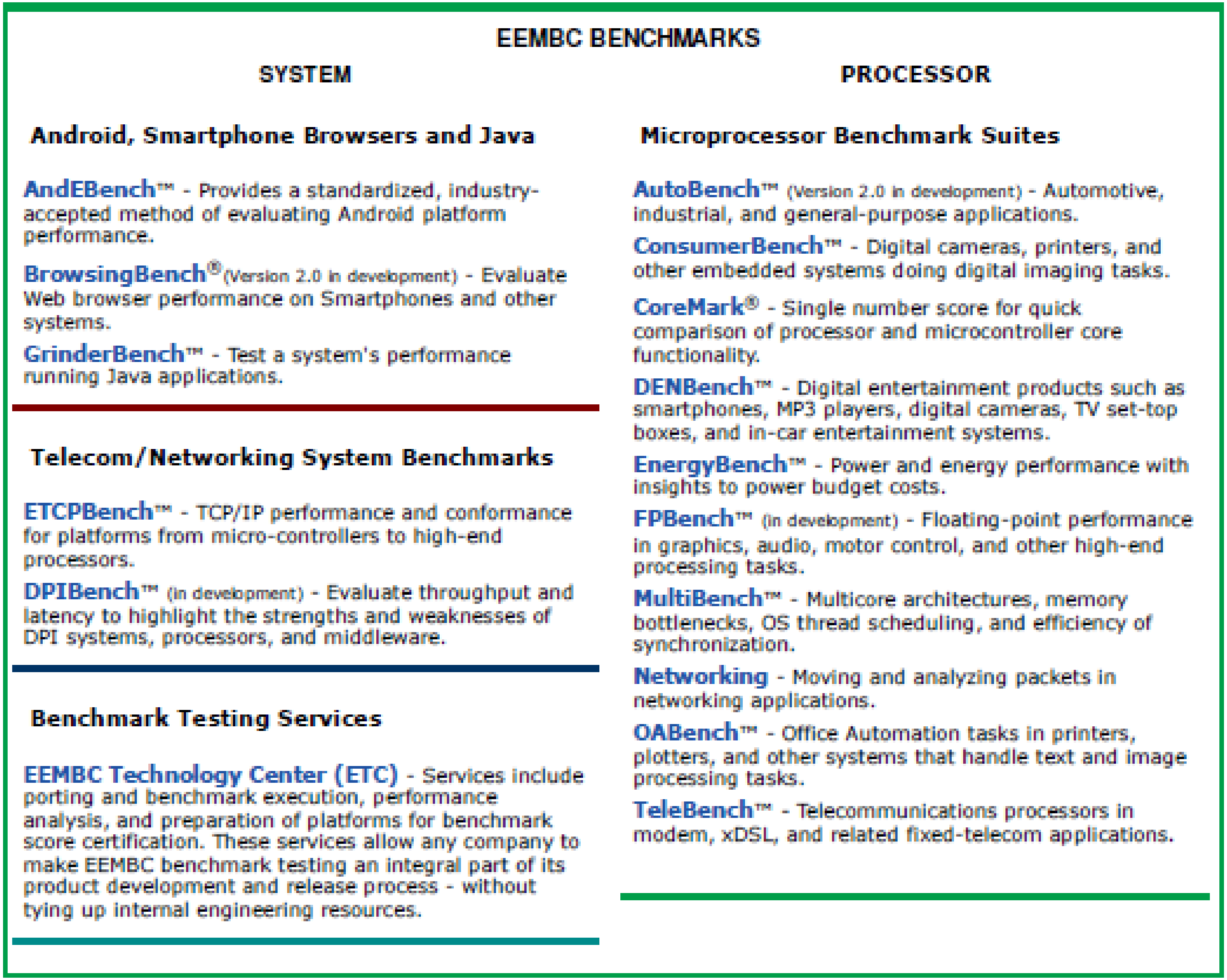
MIDP Profile

Native API Hooks

CLDC Configuration
+
Embedded Java Virtual Machine

Operating System

Device



EEMBC BENCHMARKS

SYSTEM

Android, Smartphone Browsers and Java

AndEBench™ - Provides a standardized, industry-accepted method of evaluating Android platform performance.

BrowsingBench® (Version 2.0 in development) - Evaluate Web browser performance on Smartphones and other systems.

GrinderBench™ - Test a system's performance running Java applications.

Telecom/Networking System Benchmarks

ETCPBench™ - TCP/IP performance and conformance for platforms from micro-controllers to high-end processors.

DPIBench™ (in development) - Evaluate throughput and latency to highlight the strengths and weaknesses of DPI systems, processors, and middleware.

Benchmark Testing Services

EEMBC Technology Center (ETC) - Services include porting and benchmark execution, performance analysis, and preparation of platforms for benchmark score certification. These services allow any company to make EEMBC benchmark testing an integral part of its product development and release process - without tying up internal engineering resources.

PROCESSOR

Microprocessor Benchmark Suites

AutoBench™ (Version 2.0 in development) - Automotive, industrial, and general-purpose applications.

ConsumerBench™ - Digital cameras, printers, and other embedded systems doing digital imaging tasks.

CoreMark® - Single number score for quick comparison of processor and microcontroller core functionality.

DENBench™ - Digital entertainment products such as smartphones, MP3 players, digital cameras, TV set-top boxes, and in-car entertainment systems.

EnergyBench™ - Power and energy performance with insights to power budget costs.

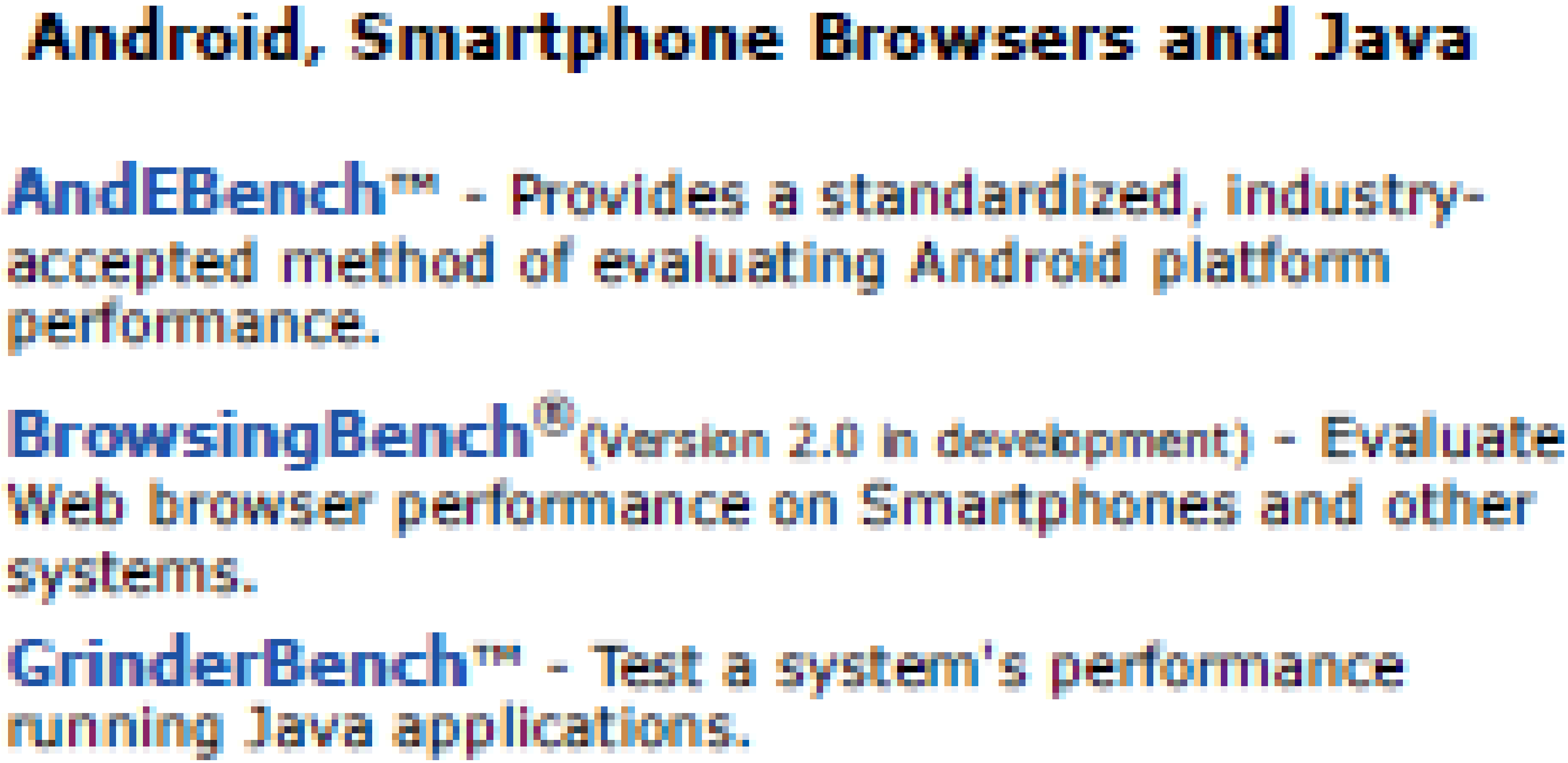
FPBench™ (in development) - Floating-point performance in graphics, audio, motor control, and other high-end processing tasks.

MultiBench™ - Multicore architectures, memory bottlenecks, OS thread scheduling, and efficiency of synchronization.

Networking - Moving and analyzing packets in networking applications.

OABench™ - Office Automation tasks in printers, plotters, and other systems that handle text and image processing tasks.

TeleBench™ - Telecommunications processors in modem, xDSL, and related fixed-telecom applications.

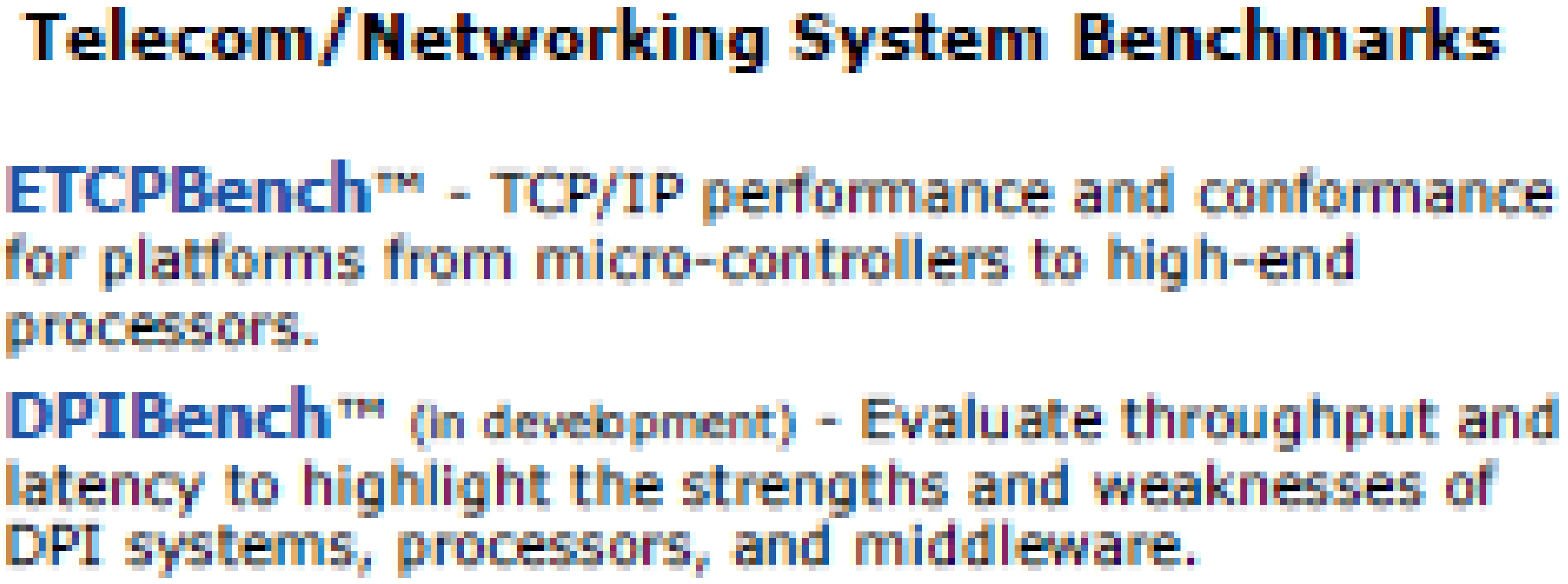


Android, Smartphone Browsers and Java

AndEBench™ - Provides a standardized, industry-accepted method of evaluating Android platform performance.

BrowsingBench® (Version 2.0 in development) - Evaluate Web browser performance on Smartphones and other systems.

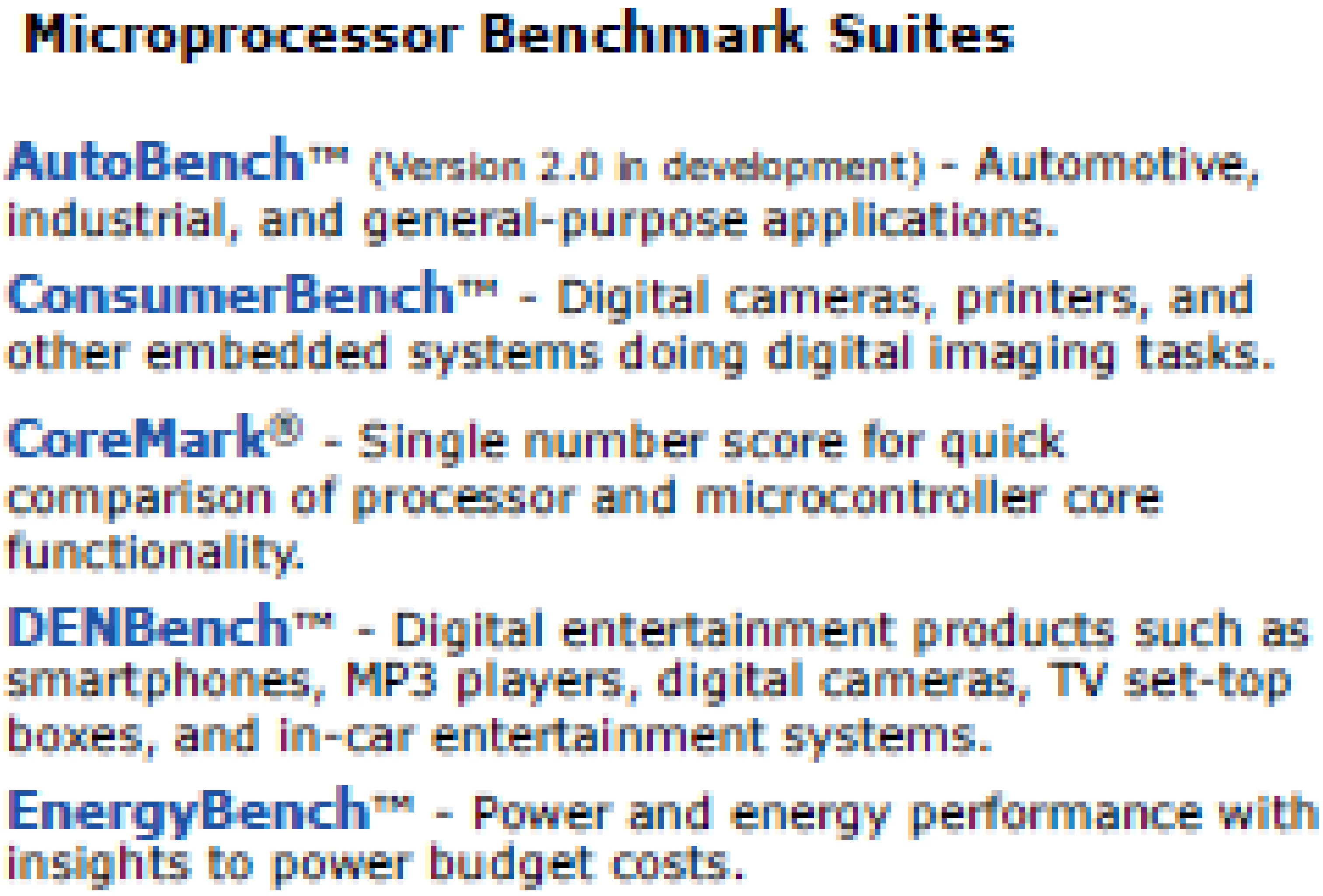
GrinderBench™ - Test a system's performance running Java applications.



Telecom/Networking System Benchmarks

ETCPBench™ - TCP/IP performance and conformance for platforms from micro-controllers to high-end processors.

DPIBench™ (in development) - Evaluate throughput and latency to highlight the strengths and weaknesses of DPI systems, processors, and middleware.



Microprocessor Benchmark Suites

AutoBench™ (Version 2.0 in development) - Automotive, industrial, and general-purpose applications.

ConsumerBench™ - Digital cameras, printers, and other embedded systems doing digital imaging tasks.

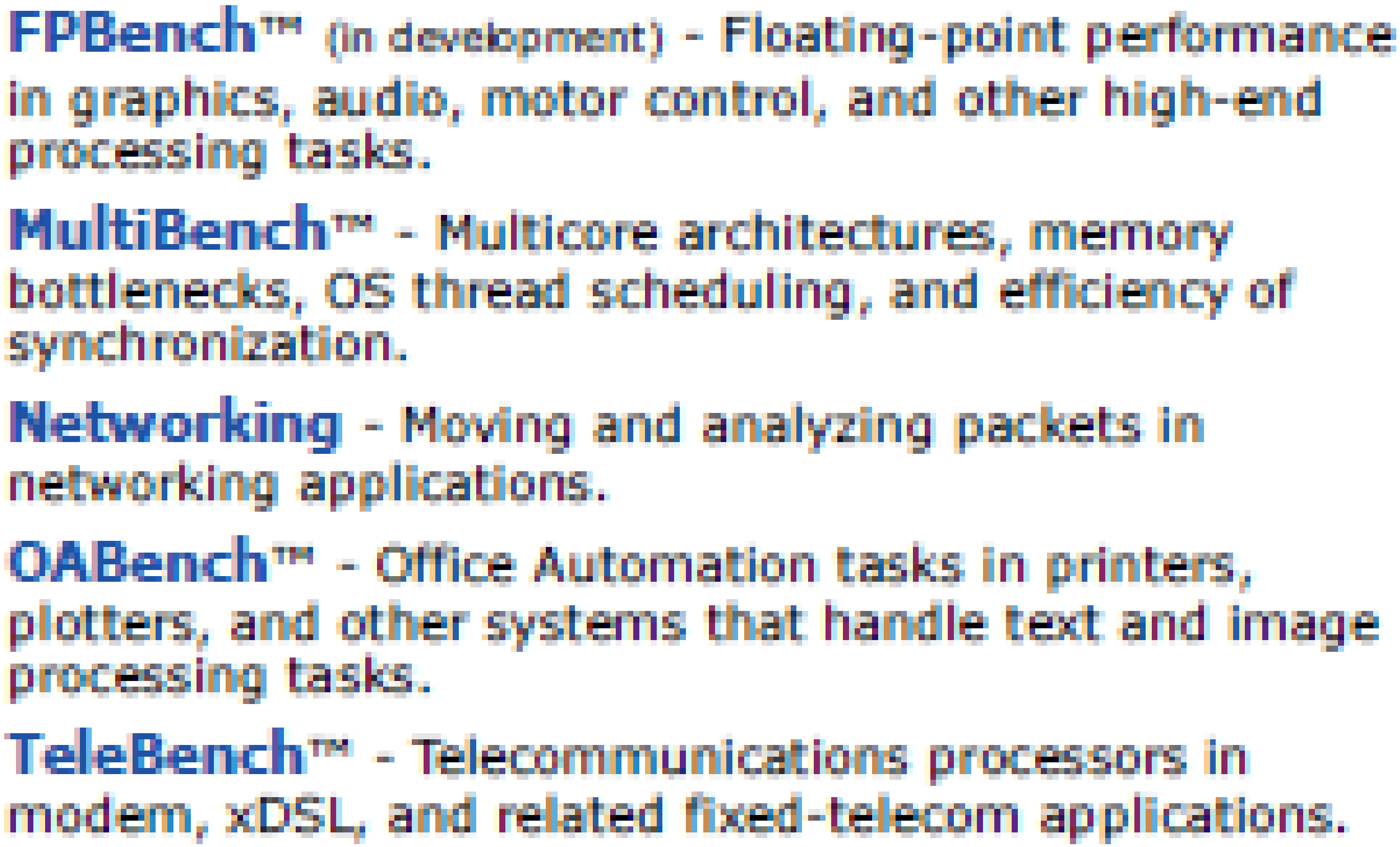
CoreMark® - Single number score for quick comparison of processor and microcontroller core functionality.

DENBench™ - Digital entertainment products such as smartphones, MP3 players, digital cameras, TV set-top boxes, and in-car entertainment systems.

EnergyBench™ - Power and energy performance with insights to power budget costs.

Coremark

- Generic Benchmark from EEMBC
- Targets the Processor Core
- Isolates CPU core from the rest of the processor
- Ignores memory and I/O effects
- Specifically targets the pipeline
- Single number score intended for quick comparison of different cores
- Influenced by Dhrystone’s simplicity



FPBench™ (in development) - Floating-point performance in graphics, audio, motor control, and other high-end processing tasks.

MultiBench™ - Multicore architectures, memory bottlenecks, OS thread scheduling, and efficiency of synchronization.

Networking - Moving and analyzing packets in networking applications.

OABench™ - Office Automation tasks in printers, plotters, and other systems that handle text and image processing tasks.

TeleBench™ - Telecommunications processors in modem, xDSL, and related fixed-telecom applications.

Calibration of Microprocessor Performance Models

Bryan Black
John P. Shen

IEEE Computer
May 1998

Sources of Error

- **Modeling error** – developer understands the modeling task but incorrectly codes the desired functionality
- **Specification error** – developer is misinformed about the correct functionality; so models the wrong functionality
- **Abstraction error** – Some features are abstracted to keep models simple/fast. The abstraction may be incorrect. Some feature may be unimplemented but has a significant impact when implemented.

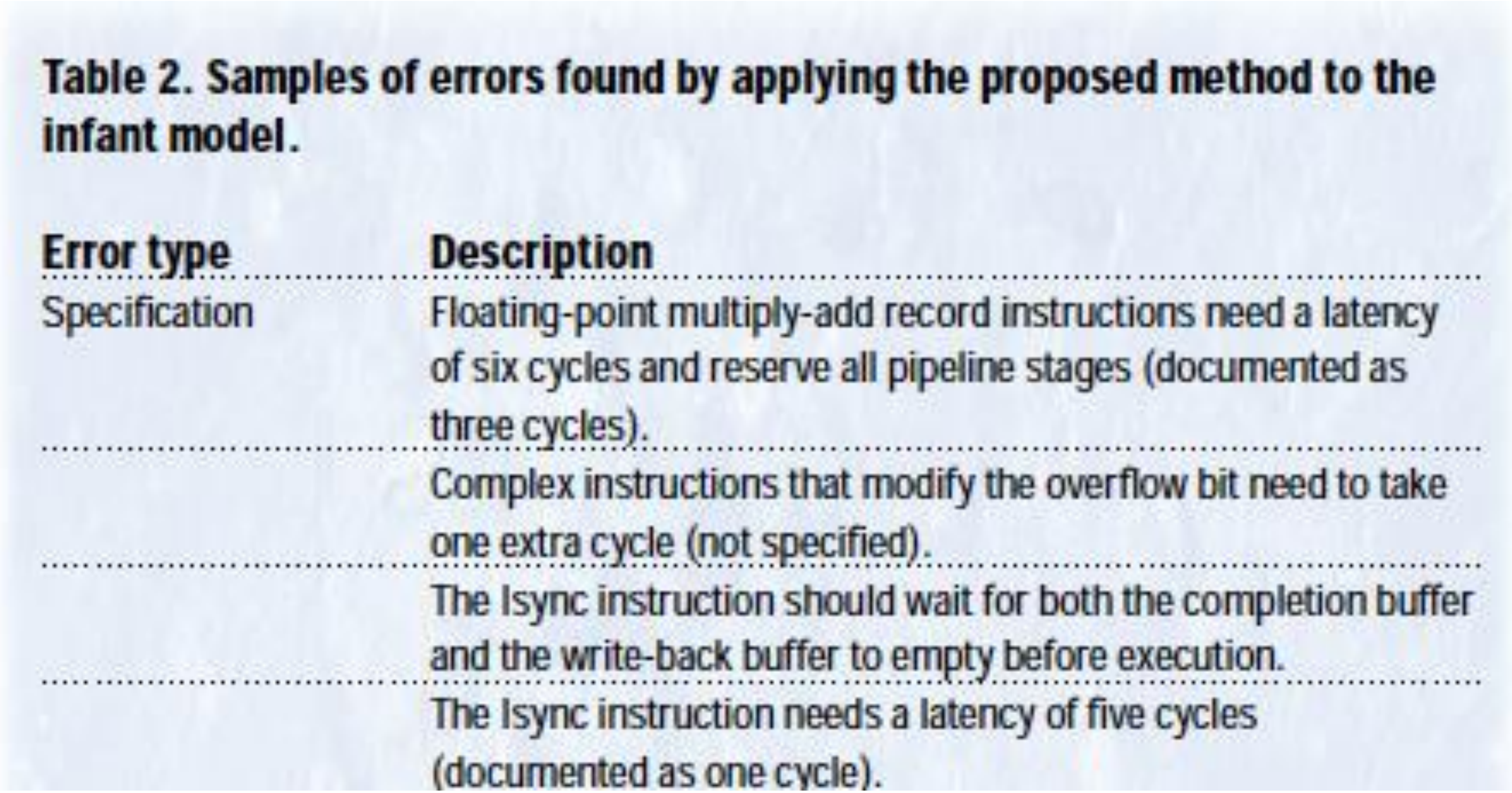
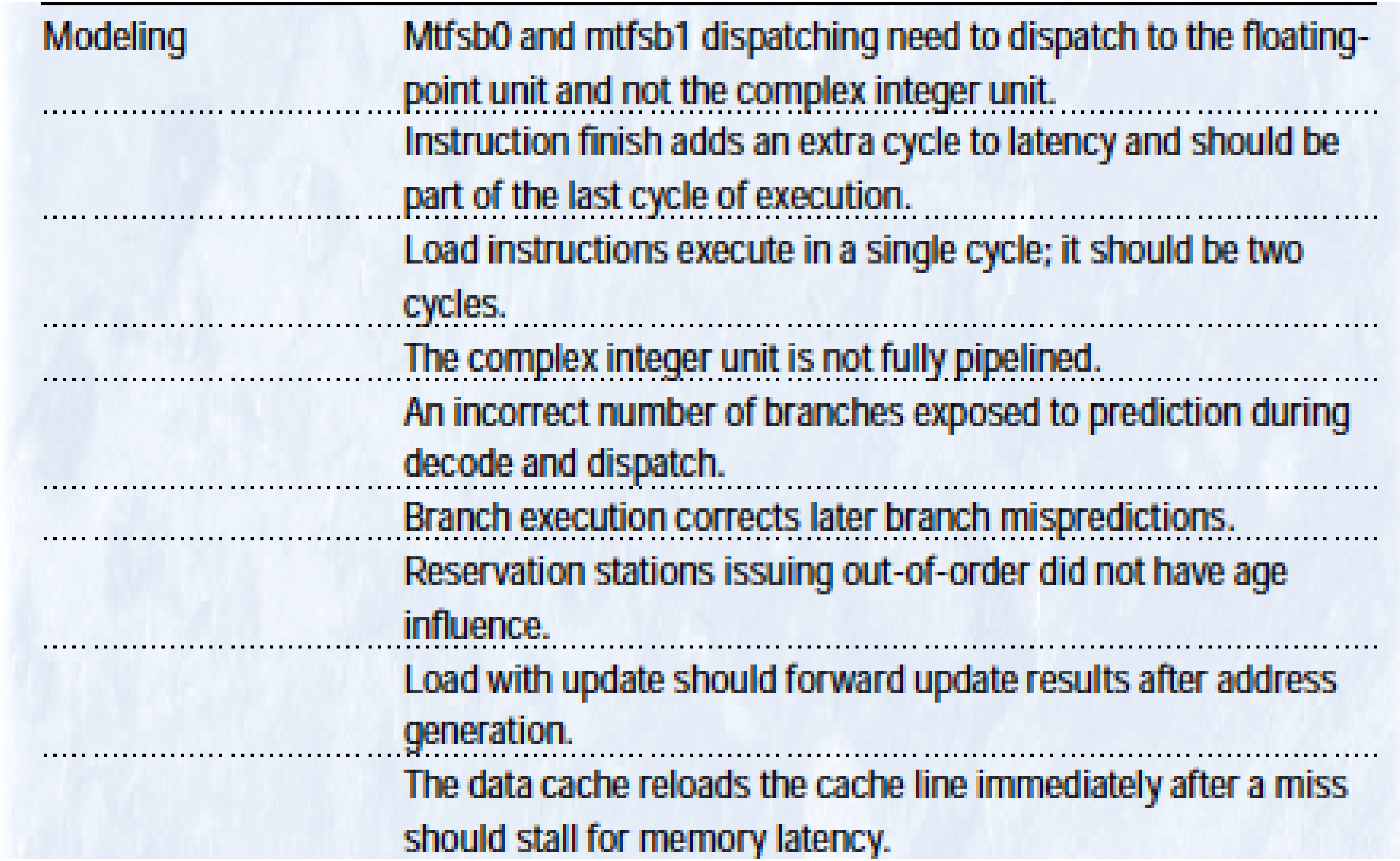


Table 2. Samples of errors found by applying the proposed method to the infant model.

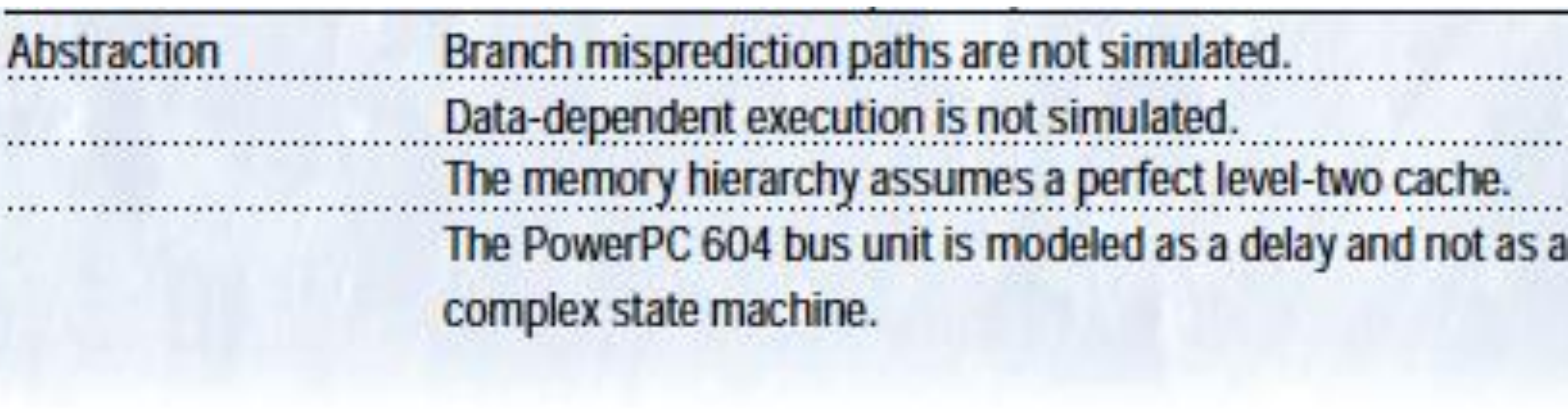
Error type	Description
Specification	Floating-point multiply-add record instructions need a latency of six cycles and reserve all pipeline stages (documented as three cycles).
	Complex instructions that modify the overflow bit need to take one extra cycle (not specified).
	The Isync instruction should wait for both the completion buffer and the write-back buffer to empty before execution.
	The Isync instruction needs a latency of five cycles (documented as one cycle).

Modeling Errors found using Infant model



Modeling	Mtfsb0 and mtfsb1 dispatching need to dispatch to the floating-point unit and not the complex integer unit.
	Instruction finish adds an extra cycle to latency and should be part of the last cycle of execution.
	Load instructions execute in a single cycle; it should be two cycles.
	The complex integer unit is not fully pipelined.
	An incorrect number of branches exposed to prediction during decode and dispatch.
	Branch execution corrects later branch mispredictions.
	Reservation stations issuing out-of-order did not have age influence.
	Load with update should forward update results after address generation.
	The data cache reloads the cache line immediately after a miss should stall for memory latency.

Abstraction Errors found using Infant model



Abstraction	Branch misprediction paths are not simulated.
	Data-dependent execution is not simulated.
	The memory hierarchy assumes a perfect level-two cache.
	The PowerPC 604 bus unit is modeled as a delay and not as a complex state machine.

Validation by Inspection

- Single-step through small code sequences
- Okay with trace-driven models
- Difficult with execution driven
- Incremental observation with incremental inputs (eg: LLL with 1000 and 2000 iterations)
- Sanity checks with an array of instrumentation counters
- Simple code sequences to exercise boundary conditions of all resources in the model

Validation Phases

- Alpha tests – exercise instruction latency by executing each instruction one at a time
- Beta tests – check pipeline dependencies with an instruction type by executing 2 to 100 instructions back to back
- Gamma tests – executes each instruction next to every other instruction type testing pipeline dependencies between instruction types

Validation Phases

- Random test sequences – upto 100 instructions exercise interactions among instructions and the different components of the microarchitecture
- ATPG – Automatic Test Pattern Generators as in digital IC testing
- Handwritten patterns – to test microarchitectural features for which tests are difficult to generate automatically

ATPG

- Alpha, beta, gamma tests are generated by ATPG. ATPG extracts ISA information from the perf model and generates executable code sequences which fall into alpha, beta, gamma tests

Case Study – Power PC Model

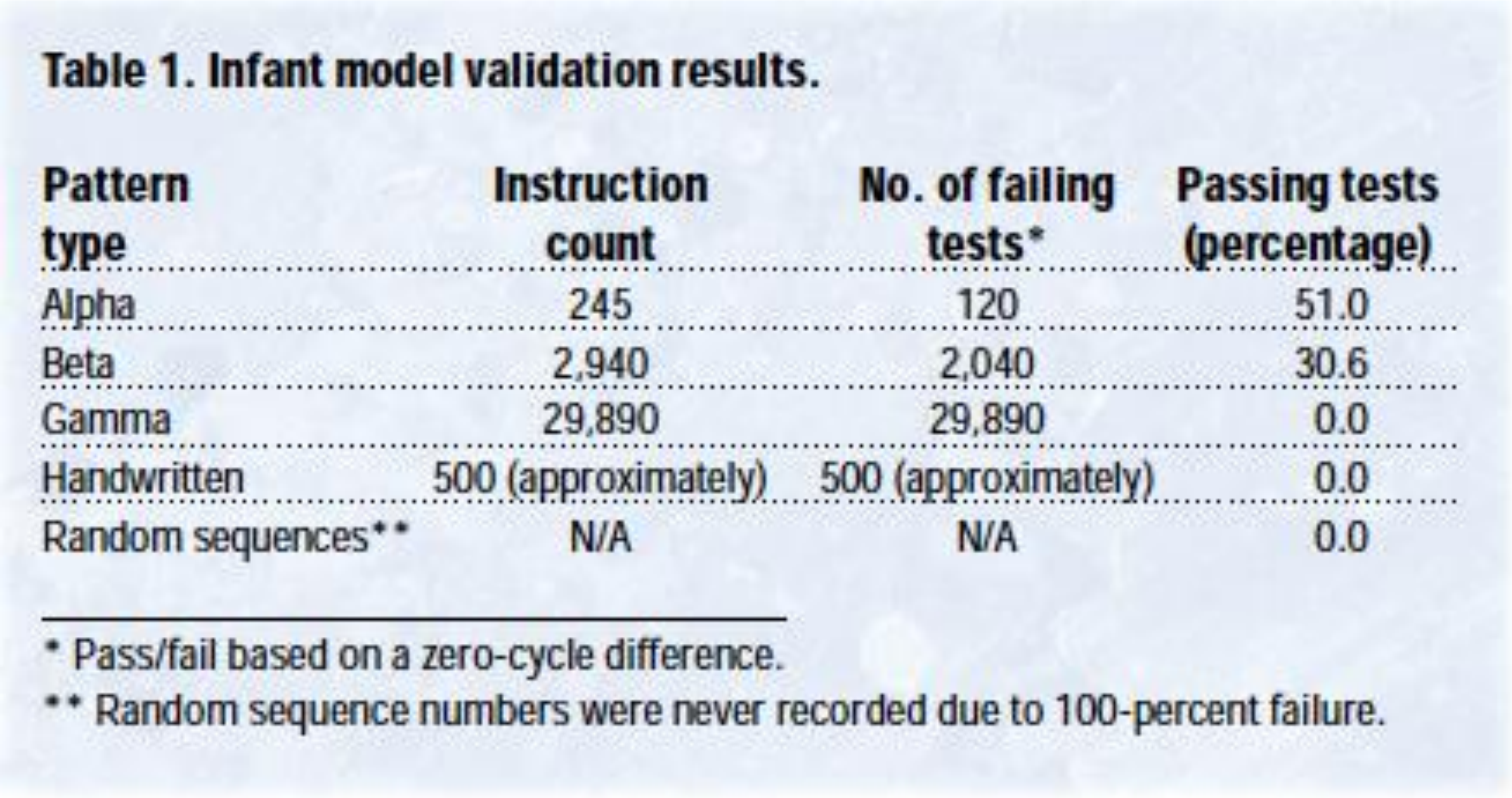


Table 1. Infant model validation results.

Pattern type	Instruction count	No. of failing tests*	Passing tests (percentage)
Alpha	245	120	51.0
Beta	2,940	2,040	30.6
Gamma	29,890	29,890	0.0
Handwritten	500 (approximately)	500 (approximately)	0.0
Random sequences**	N/A	N/A	0.0

* Pass/fail based on a zero-cycle difference.

** Random sequence numbers were never recorded due to 100-percent failure.

Table 3. Child model validation results.

Pattern type	Instruction count	Zero-cycle difference		One-cycle difference	
		No. of failing tests	Passing tests (percentage)	No. of failing tests	Passing tests (percentage)
Alpha	194	8	95.9	0	100.0
Beta	20,358	5,017	75.4	3,105	84.7
Gamma	60,552	12,244	79.8	11,005	81.8
Handwritten	about 500	0	100.0	Not applicable	Not applicable
Random sequences	93,779	63,782	32.0	26,688	71.5

I-count discrepancy

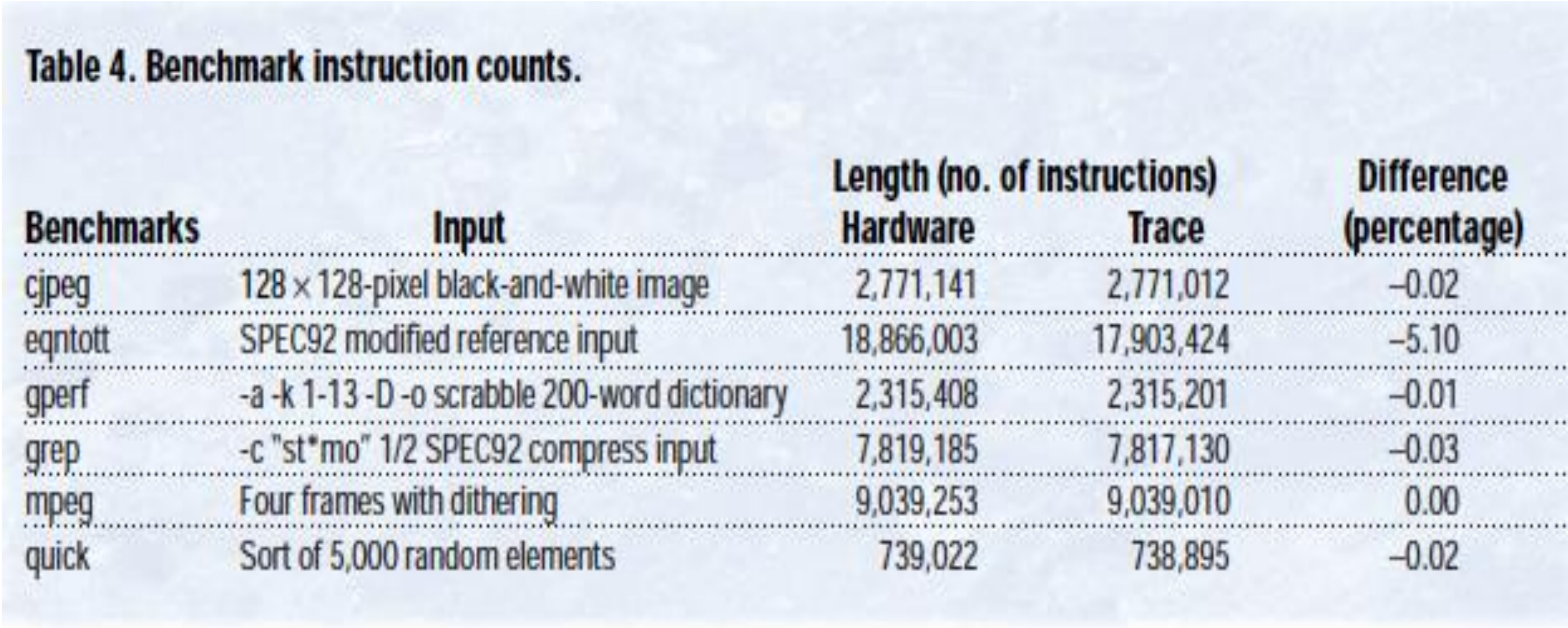


Table 4. Benchmark instruction counts.

Benchmarks	Input	Length (no. of instructions) Hardware	 Trace	Difference (percentage)
cjpeg	128 × 128-pixel black-and-white image	2,771,141	2,771,012	–0.02
eqntott	SPEC92 modified reference input	18,866,003	17,903,424	–5.10
gperf	-a -k 1-13 -D -o scrabble 200-word dictionary	2,315,408	2,315,201	–0.01
grep	-c "st*mo" 1/2 SPEC92 compress input	7,819,185	7,817,130	–0.03
mpeg	Four frames with dithering	9,039,253	9,039,010	0.00
quick	Sort of 5,000 random elements	739,022	738,895	–0.02

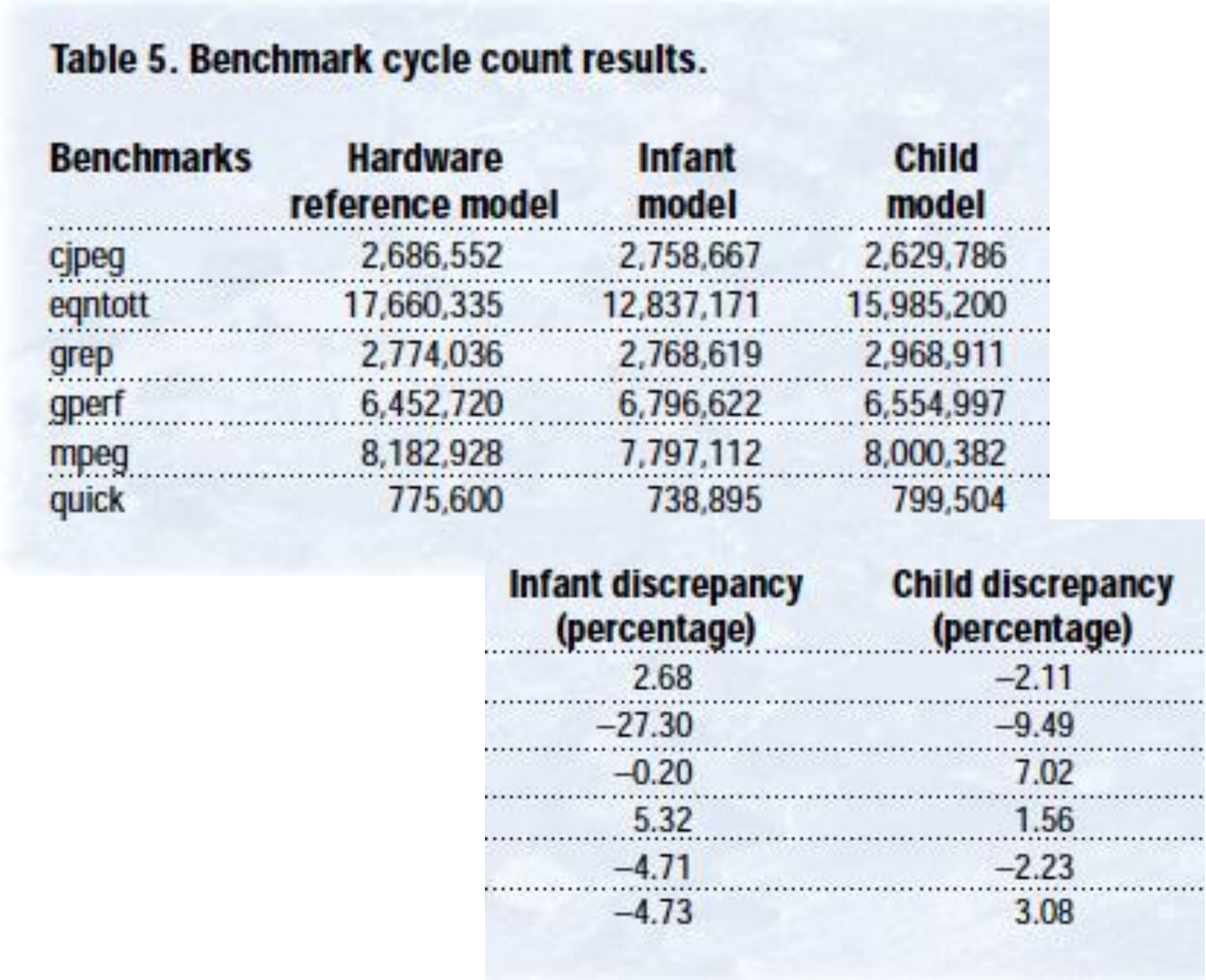


Table 5. Benchmark cycle count results.

Benchmarks	Hardware reference model	Infant model	Child model	Infant discrepancy (percentage)	Child discrepancy (percentage)
cjpeg	2,686,552	2,758,667	2,629,786	2.68	–2.11
eqntott	17,660,335	12,837,171	15,985,200	–27.30	–9.49
grep	2,774,036	2,768,619	2,968,911	–0.20	7.02
gperf	6,452,720	6,796,622	6,554,997	5.32	1.56
mpeg	8,182,928	7,797,112	8,000,382	–4.71	–2.23
quick	775,600	738,895	799,504	–4.73	3.08

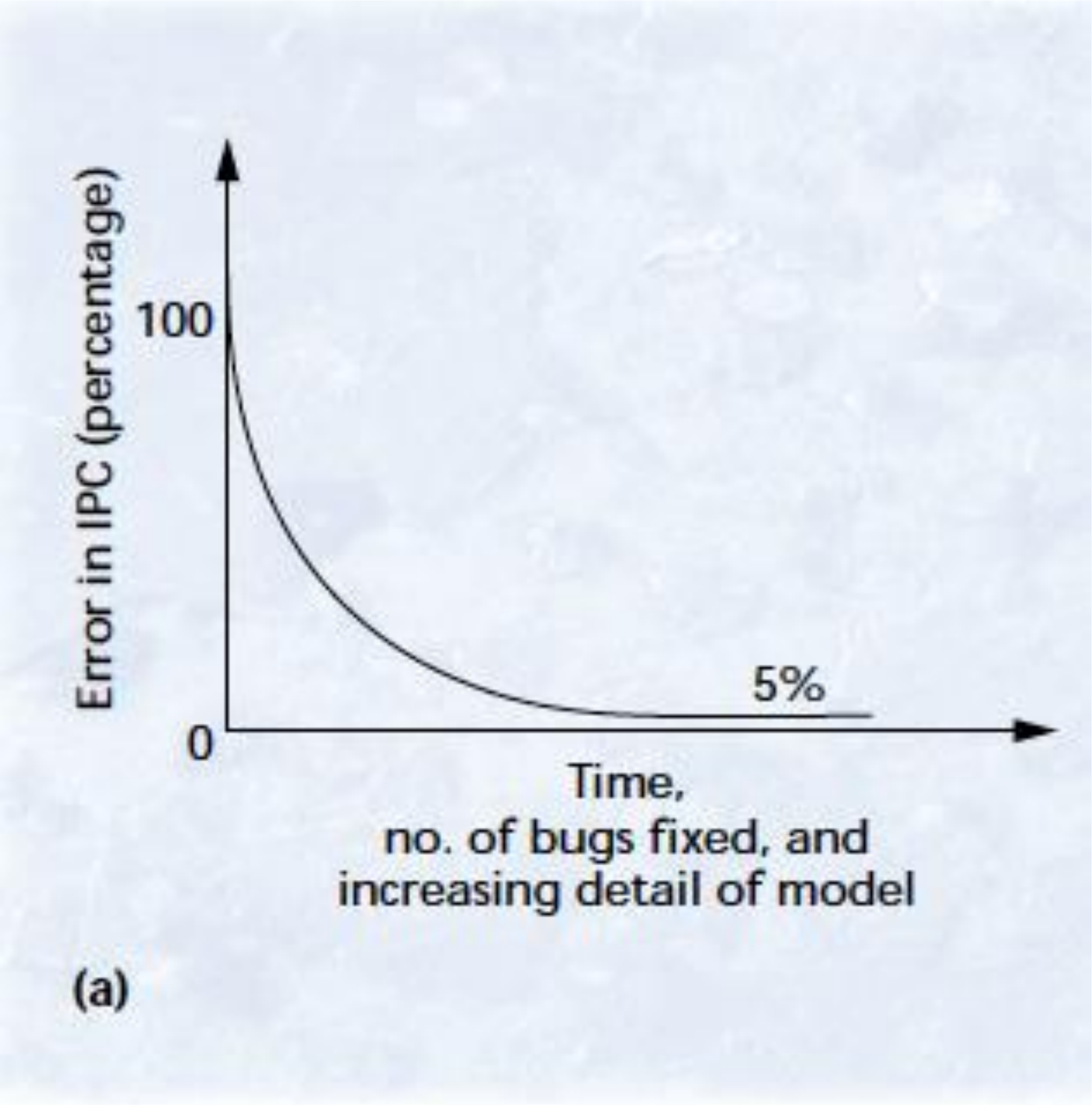


Figure 2. Conflicting views of the maturation of a performance model: (a) simplistic view in which inaccuracies decrease monotonically and (b) realistic view in which results oscillate and then slowly converge.

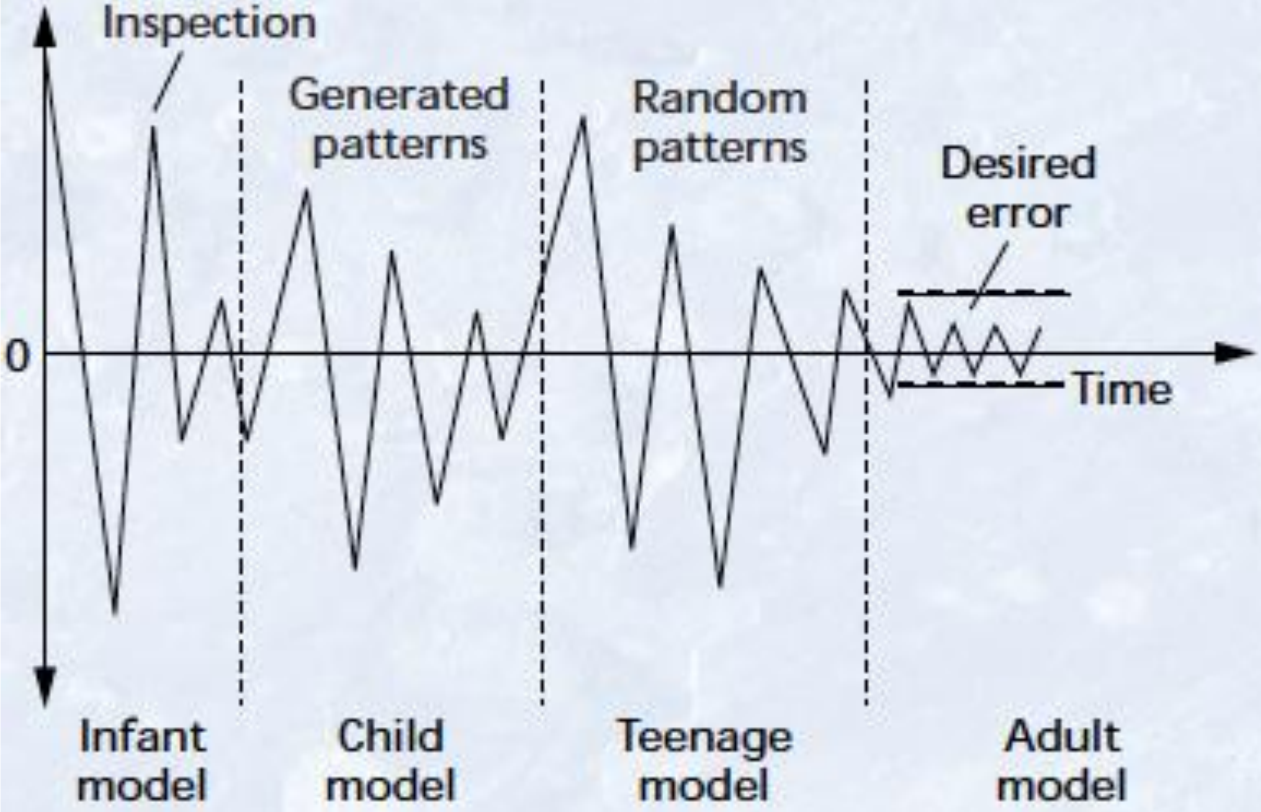


Figure 2. Conflicting views of the maturation of a performance model: (a) simplistic view in which inaccuracies decrease monotonically and (b) realistic view in which results oscillate and then slowly converge.