CEP4Cloud - Complex Event Processing for Mastering the Elastic Cloud

Distributed Systems Group
Database Research Group
University of Marburg, Germany

Software AG - ProcessWorld 2011

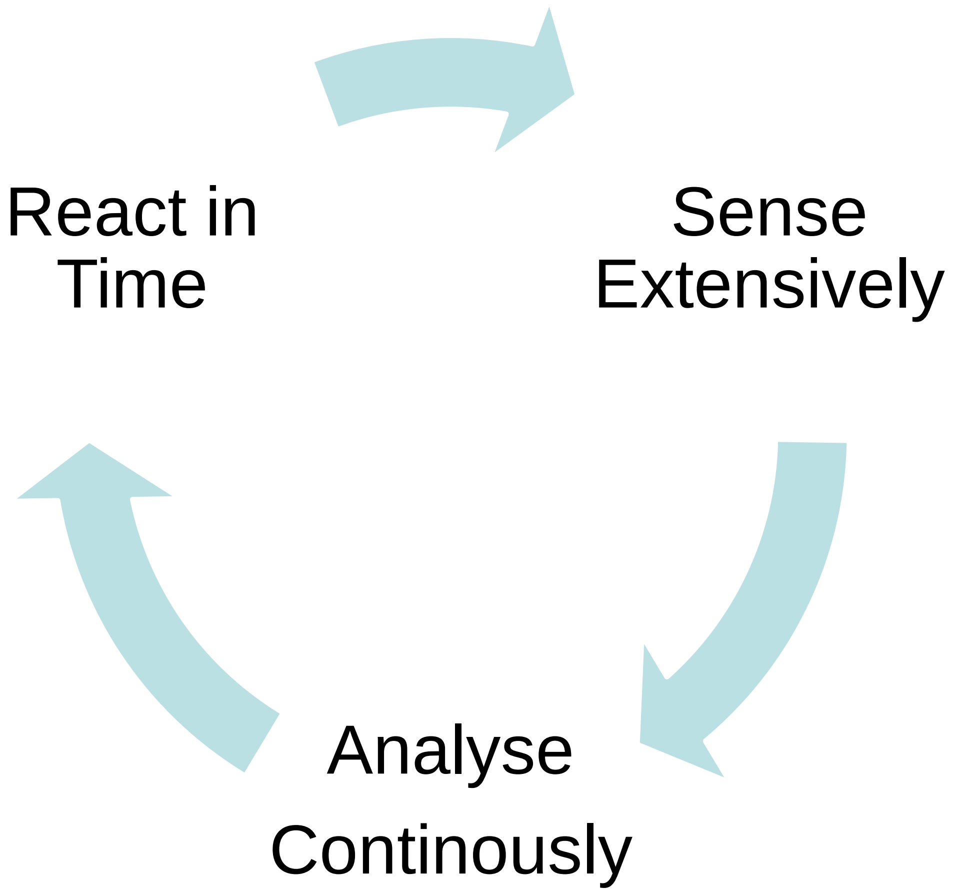
Elastic Cloud Computing

- Gartner clients' No. 1 topic in 2010

  “A style of computing where **scalable** and **elastic** IT-enabled capabilities are delivered as a **service** to **customers** using Internet technologies”

- Five essential characteristics (NIST)
  - **Resource Pooling**
  - **Rapid Elasticity**
  - **Measured Service**
- The Challenge
  - Guarantee of Service Level Agreements in a Highly Dynamic World of Elastic IT-Resources and Services

CEP4Cloud



React in Time

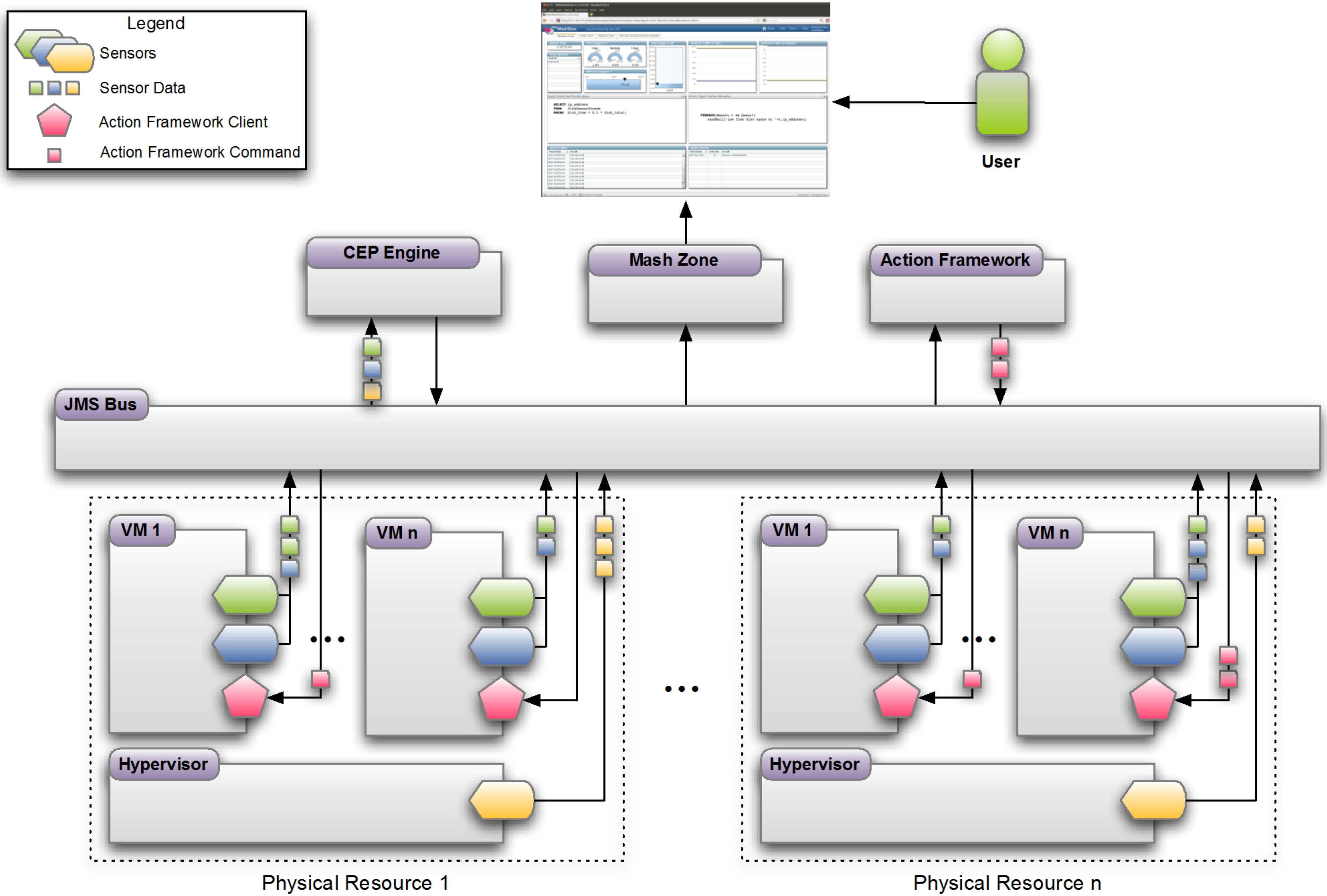
Sense Extensively

Analyse Continously

CEP4Cloud: Architecture

- Sensors running on different cloud layers
  - Application, Hardware, System (Linux, Windows)
- Easy activation/deactivation using a bundle framework
  - Sensors load dependencies automatically
  - Provides messaging infrastructure (JMS, AMQP)
- CEP engine queries the data streams
  - Continuously observation, filtering and combination of streams
  - Extraction of relevant information from input data
- Aris MashZone is used to show the results
- Action Framework executes custom actions

CEP4Cloud: Architecture



```
SELECT ip_address
FROM   DiskSensorStream
WHERE  disk_free < 0.2 * disk_total;
```

```
FOREACH(Result t in Query1)
  sendMail('Low free disk space at '+t.ip_address);
```

CEP4Cloud: CEP Engine

- New and powerful monitoring possibilities
- Multi-dimensional cross-layer analyses
- Queries are expressed declarative in SQL
- Pattern matching on single and combined streams
- Detection of trends, evolutions and abnormal situations
- Scalable

CEP4Cloud: Action Framework

- Listens on JMS bus for incoming events
- Executes actions on cloud instances
  - Custom JavaScript code
- Actions
  - Notify admin via email
  - Execute actions on system and application layer
- Has access to sensor's interfaces and data

Contact

Distributed Systems Group
Prof. Dr. Bernd Freisleben
freisleb@informatik.uni-marburg.de
Hans-Meerwein-Strasse 3
35032 Marburg
Fon: +49 6421 / 28 21 568
Fax: +49 6421 / 28 21 573

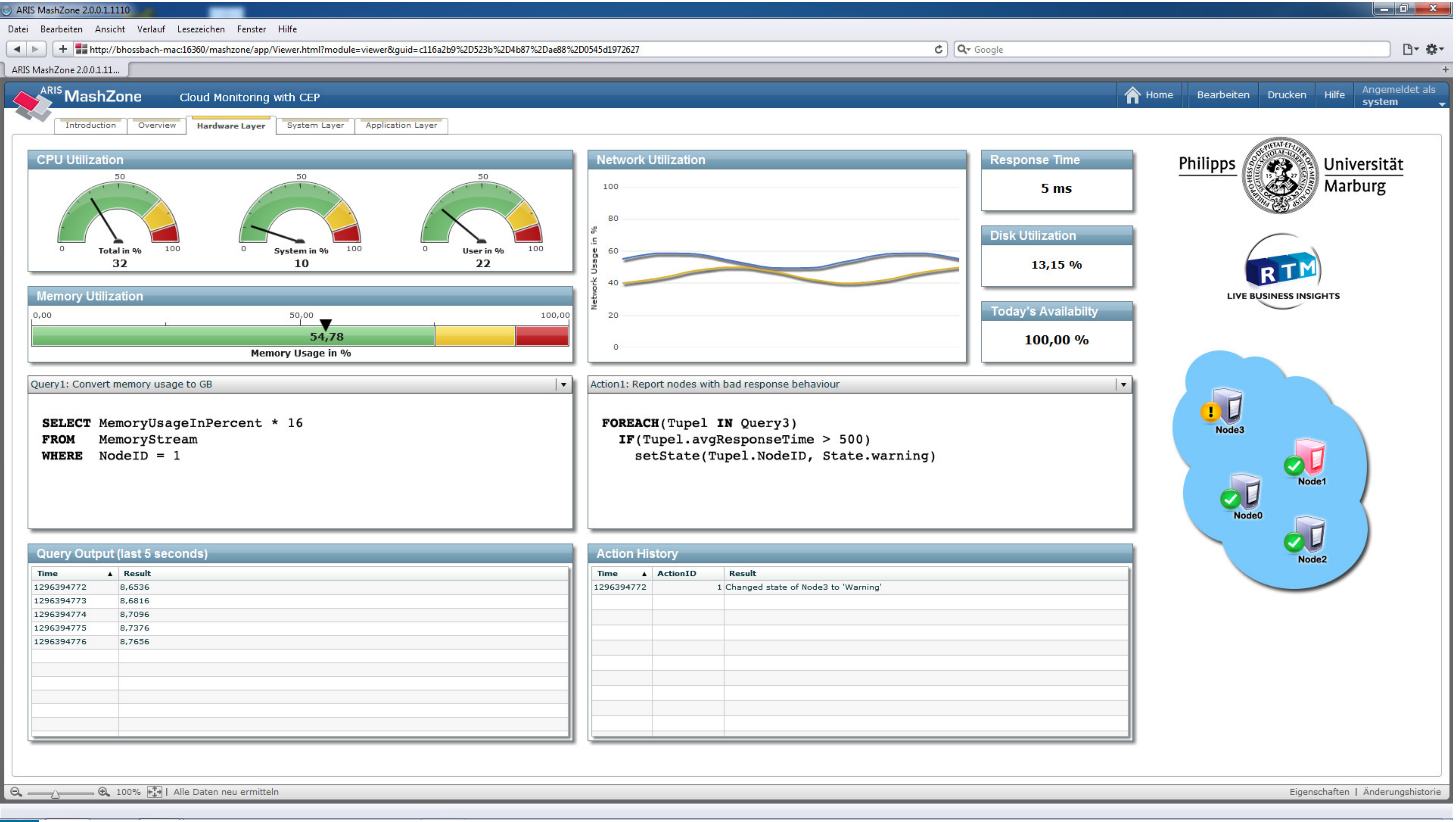
Database Research Group
Prof. Dr. Bernhard Seeger
seeger@informatik.uni-marburg.de
Hans-Meerwein-Strasse 3
35032 Marburg
Fon: +49 6421 / 28 21 526
Fax: +49 6421 / 28 21 573

Team

Lars Baumgärtner, Pablo Graubner, Bastian Hoßbach,
Matthias Schmidt, Roland Schwarzkopf

BACKUP SLIDES

Hardware Layer



Response Time

5 ms

Disk Utilization

13,15 %

Today's Availabilty

100,00 %

Query1: Convert memory usage to GB

```
SELECT MemoryUsageInPercent * 16
FROM   MemoryStream
WHERE  NodeID = 1
```

Action1: Report nodes with bad response behaviour

```
FOREACH(Tupel IN Query3)
  IF(Tupel.avgResponseTime > 500)
    setState(Tupel.NodeID, State.warning)
```

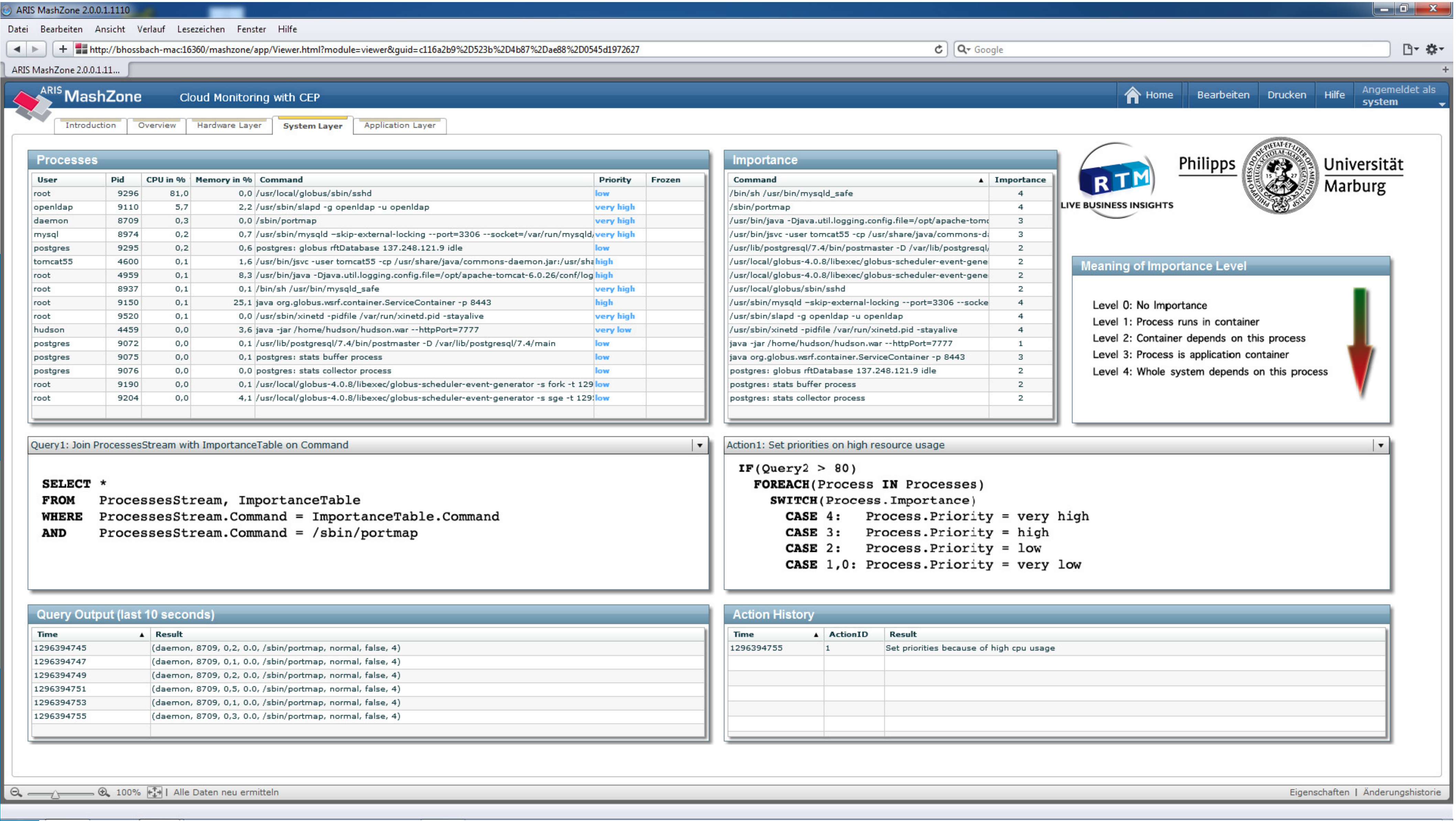
Query Output (last 5 seconds)

Time	Result
1296394772	8,6536
1296394773	8,6816
1296394774	8,7096
1296394775	8,7376
1296394776	8,7656

Action History

Time	ActionID	Result
1296394772	1	Changed state of Node3 to 'Warning'

System Layer


ARIS MashZone 2.0.0.1.1110
Datei Bearbeiten Ansicht Verlauf Lesezeichen Fenster Hilfe
http://bhossbach-mac:16360/mashzone/app/Viewer.html?module=viewer&guid=c116a2b9%2D523b%2D4b87%2Dae88%2D0545d1972627
Google
ARIS MashZone 2.0.0.1.11...
ARIS MashZone
Cloud Monitoring with CEP
Home
Bearbeiten
Drucken
Hilfe
Angemeldet als system
Introduction
Overview
Hardware Layer
System Layer
Application Layer
RTM LIVE BUSINESS INSIGHTS
Philipps Universität Marburg
100%
Alle Daten neu ermitteln
Eigenschaften | Änderungshistorie

Processes

User	Pid	CPU in %	Memory in %	Command	Priority	Frozen
root	9296	81,0	0,0	/usr/local/globus/sbin/sshd	low	
openldap	9110	5,7	2,2	/usr/sbin/slapd -g openldap -u openldap	very high	
daemon	8709	0,3	0,0	/sbin/portmap	very high	
mysql	8974	0,2	0,7	/usr/sbin/mysqld –skip-external-locking --port=3306 --socket=/var/run/mysqld/	very high	
postgres	9295	0,2	0,6	postgres: globus rftDatabase 137.248.121.9 idle	low	
tomcat55	4600	0,1	1,6	/usr/bin/jsvc -user tomcat55 -cp /usr/share/java/commons-daemon.jar:/usr/sha	high	
root	4959	0,1	8,3	/usr/bin/java -Djava.util.logging.config.file=/opt/apache-tomcat-6.0.26/conf/log	high	
root	8937	0,1	0,1	/bin/sh /usr/bin/mysqld_safe	very high	
root	9150	0,1	25,1	java org.globus.wsrf.container.ServiceContainer -p 8443	high	
root	9520	0,1	0,0	/usr/sbin/xinetd -pidfile /var/run/xinetd.pid -stayalive	very high	
hudson	4459	0,0	3,6	java -jar /home/hudson/hudson.war --httpPort=7777	very low	
postgres	9072	0,0	0,1	/usr/lib/postgresql/7.4/bin/postmaster -D /var/lib/postgresql/7.4/main	low	
postgres	9075	0,0	0,1	postgres: stats buffer process	low	
postgres	9076	0,0	0,0	postgres: stats collector process	low	
root	9190	0,0	0,1	/usr/local/globus-4.0.8/libexec/globus-scheduler-event-generator -s fork -t 129	low	
root	9204	0,0	4,1	/usr/local/globus-4.0.8/libexec/globus-scheduler-event-generator -s sge -t 129:	low	

Importance

Command ▲	Importance
/bin/sh /usr/bin/mysqld_safe	4
/sbin/portmap	4
/usr/bin/java -Djava.util.logging.config.file=/opt/apache-tomc	3
/usr/bin/jsvc -user tomcat55 -cp /usr/share/java/commons-da	3
/usr/lib/postgresql/7.4/bin/postmaster -D /var/lib/postgresql/	2
/usr/local/globus-4.0.8/libexec/globus-scheduler-event-gene	2
/usr/local/globus-4.0.8/libexec/globus-scheduler-event-gene	2
/usr/local/globus/sbin/sshd	2
/usr/sbin/mysqld –skip-external-locking --port=3306 --socke	4
/usr/sbin/slapd -g openldap -u openldap	4
/usr/sbin/xinetd -pidfile /var/run/xinetd.pid -stayalive	4
java -jar /home/hudson/hudson.war --httpPort=7777	1
java org.globus.wsrf.container.ServiceContainer -p 8443	3
postgres: globus rftDatabase 137.248.121.9 idle	2
postgres: stats buffer process	2
postgres: stats collector process	2

Meaning of Importance Level

Level 0: No Importance
Level 1: Process runs in container
Level 2: Container depends on this process
Level 3: Process is application container
Level 4: Whole system depends on this process

Query1: Join ProcessesStream with ImportanceTable on Command

```
SELECT *
FROM   ProcessesStream, ImportanceTable
WHERE  ProcessesStream.Command = ImportanceTable.Command
AND    ProcessesStream.Command = /sbin/portmap
```

Action1: Set priorities on high resource usage

```
IF(Query2 > 80)
  FOREACH(Process IN Processes)
    SWITCH(Process.Importance)
      CASE 4:   Process.Priority = very high
      CASE 3:   Process.Priority = high
      CASE 2:   Process.Priority = low
      CASE 1,0: Process.Priority = very low
```

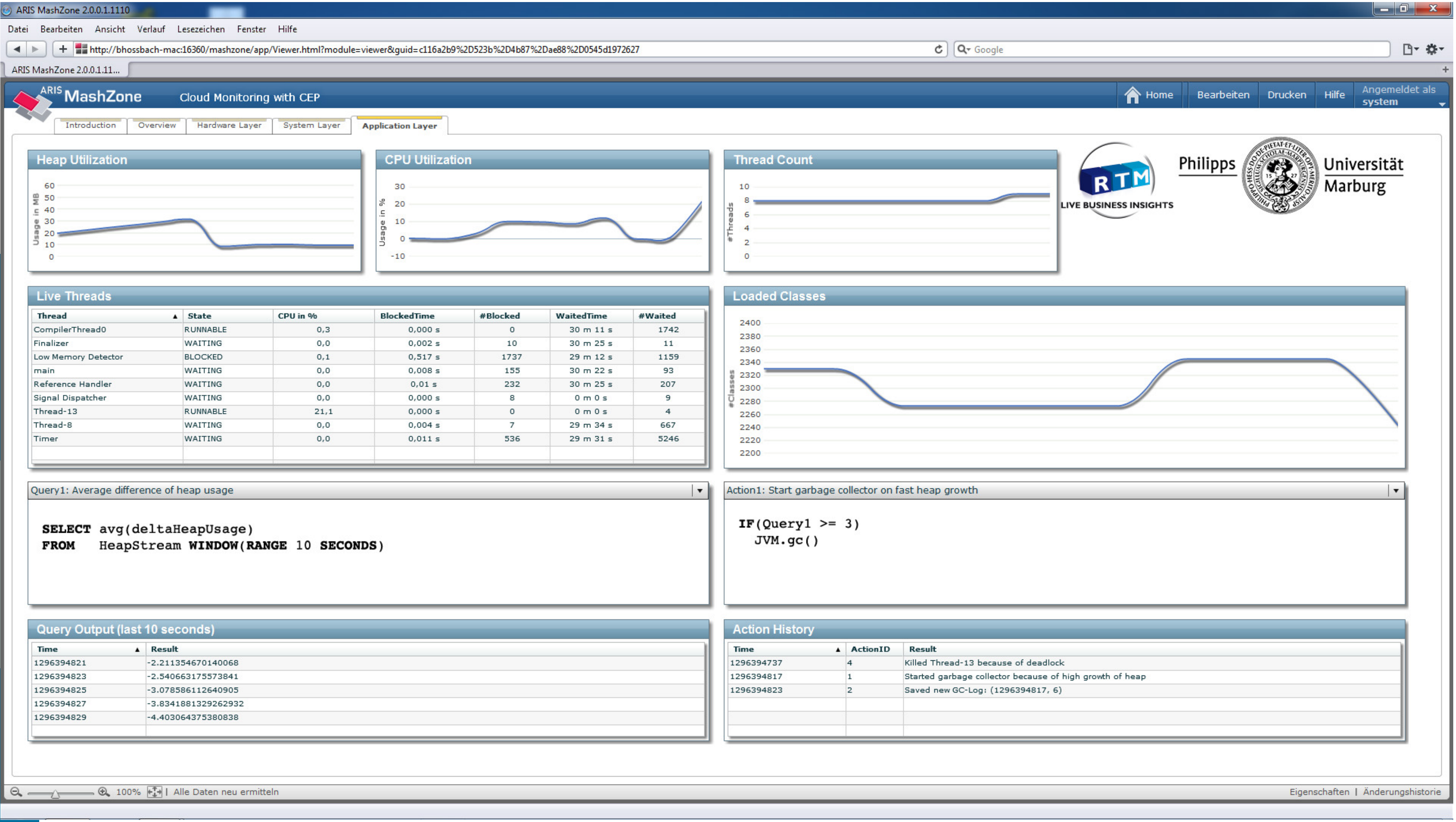
Query Output (last 10 seconds)

Time ▲	Result
1296394745	(daemon, 8709, 0,2, 0.0, /sbin/portmap, normal, false, 4)
1296394747	(daemon, 8709, 0,1, 0.0, /sbin/portmap, normal, false, 4)
1296394749	(daemon, 8709, 0,2, 0.0, /sbin/portmap, normal, false, 4)
1296394751	(daemon, 8709, 0,5, 0.0, /sbin/portmap, normal, false, 4)
1296394753	(daemon, 8709, 0,1, 0.0, /sbin/portmap, normal, false, 4)
1296394755	(daemon, 8709, 0,3, 0.0, /sbin/portmap, normal, false, 4)

Action History

Time ▲	ActionID	Result
1296394755	1	Set priorities because of high cpu usage

Application Layer



Live Threads

Thread ▲	State	CPU in %	BlockedTime	#Blocked	WaitedTime	#Waited
CompilerThread0	RUNNABLE	0,3	0,000 s	0	30 m 11 s	1742
Finalizer	WAITING	0,0	0,002 s	10	30 m 25 s	11
Low Memory Detector	BLOCKED	0,1	0,517 s	1737	29 m 12 s	1159
main	WAITING	0,0	0,008 s	155	30 m 22 s	93
Reference Handler	WAITING	0,0	0,01 s	232	30 m 25 s	207
Signal Dispatcher	WAITING	0,0	0,000 s	8	0 m 0 s	9
Thread-13	RUNNABLE	21,1	0,000 s	0	0 m 0 s	4
Thread-8	WAITING	0,0	0,004 s	7	29 m 34 s	667
Timer	WAITING	0,0	0,011 s	536	29 m 31 s	5246

Query1: Average difference of heap usage ▾

```
SELECT avg(deltaHeapUsage)
FROM   HeapStream WINDOW(RANGE 10 SECONDS)
```

Action1: Start garbage collector on fast heap growth ▾

```
IF(Query1 >= 3)
  JVM.gc()
```

Query Output (last 10 seconds)

Time ▲	Result
1296394821	-2.211354670140068
1296394823	-2.540663175573841
1296394825	-3.078586112640905
1296394827	-3.8341881329262932
1296394829	-4.403064375380838

Action History

Time ▲	ActionID	Result
1296394737	4	Killed Thread-13 because of deadlock
1296394817	1	Started garbage collector because of high growth of heap
1296394823	2	Saved new GC-Log: (1296394817, 6)