



DGUV

Deutsche Gesetzliche Unfallversicherung
Spitzenverband

Von AQ auf WebLogic JMS

Motive und Hintergründe einer Entscheidung

Rossen Rachev, DOAG 2013 Development

26.06.2013

Agenda

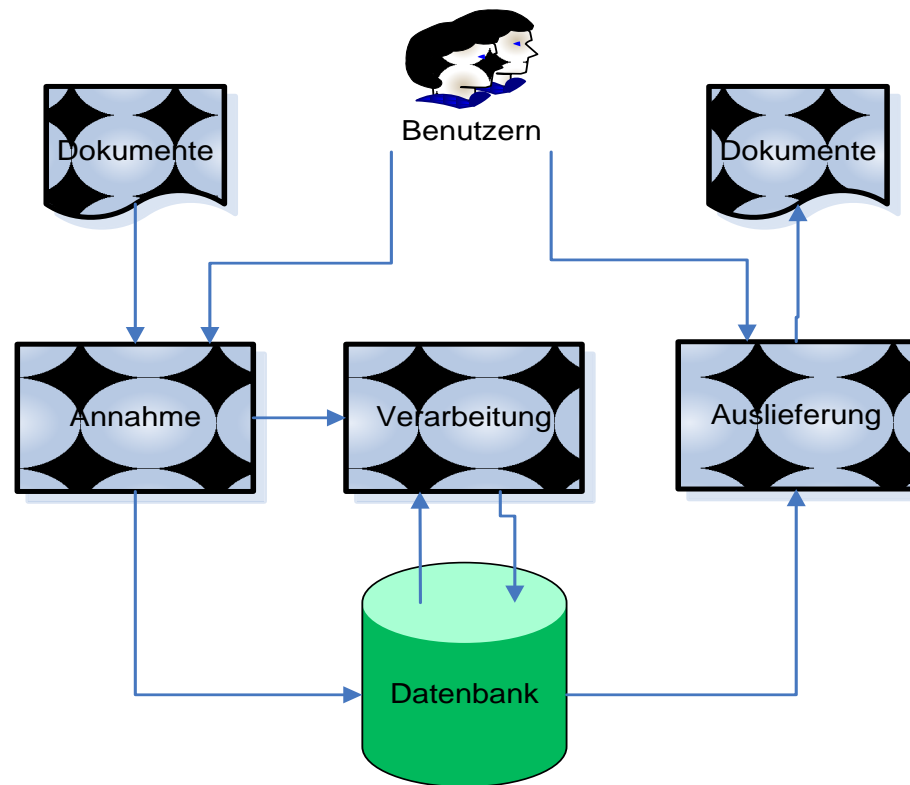
- Einführung
- Anwendungsumfeld
- Anwendungsmodell und Hosting-Infrastruktur
- Motiven für die Suche nach Alternativen (JMS Provider)
- Konzepte und Misskonzepte über das WebLogic JMS
- Auf der Suche nach eine Ausfallsichere Lösung
- Lizenzkosten-Überlegungen
- Zusammenfassung
- Fragen, Ideen, Vorschläge

Anwendungsumfeld

- DGUV – Deutsche Gesetzliche Unfallversicherung (Berufsgenossenschaften / Unfallkassen / Landesverbände)
- Betrieb von unterschiedlichen Datenaustauschverfahren
- Über 100 000 Dokumente pro Tag
- Zwei Rechenzentren
- Zwei DBAs
- Oracle Datenbank und Oracle Application Server (OAS/WLS)

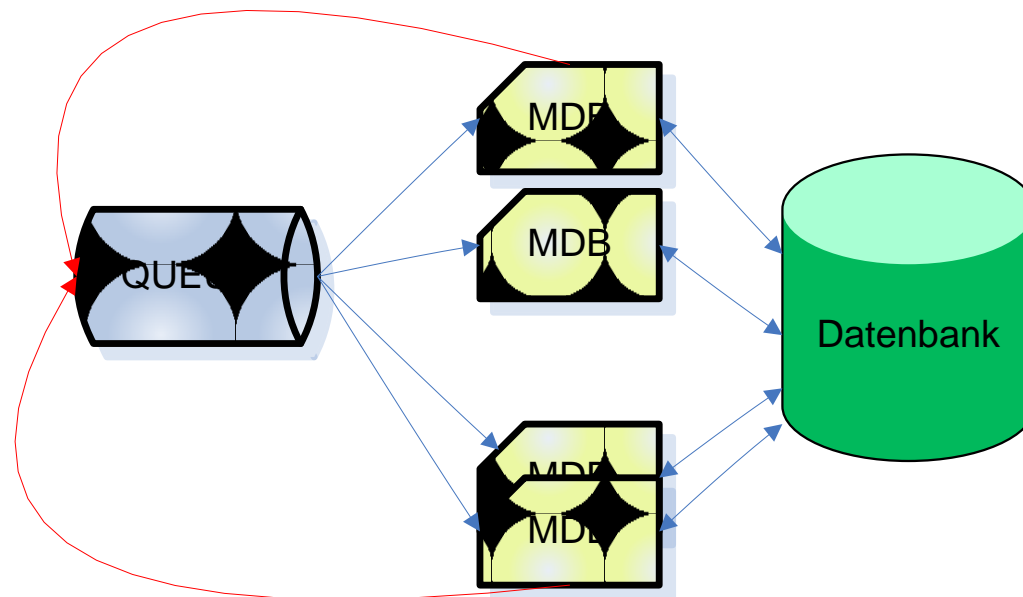
Anwendungsmodell und Hosting-Infrastruktur

Anwendungsmodell



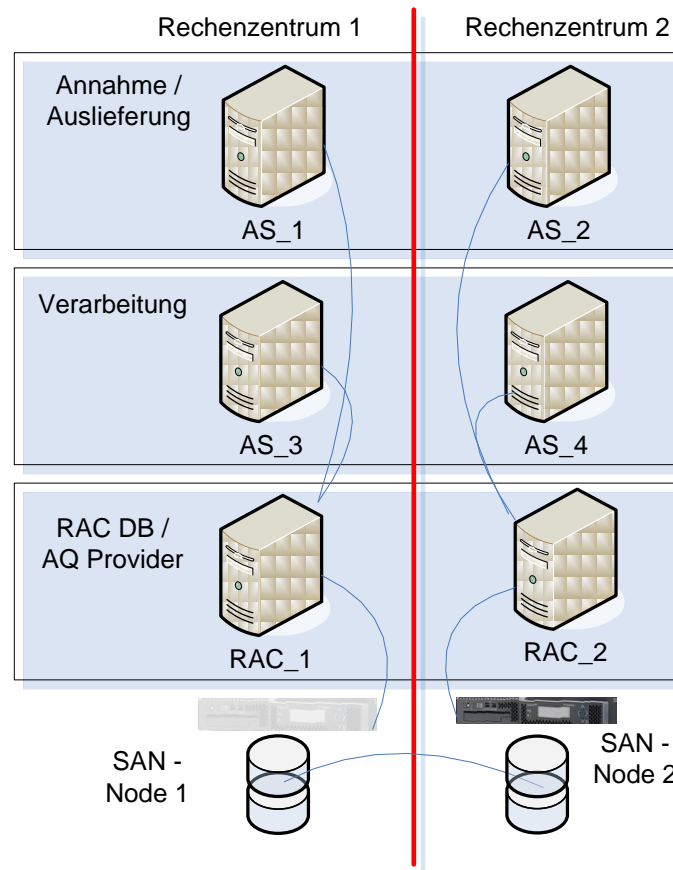
Anwendungsmodell

Verarbeitung



Anwendungsmodell und Hosting-Infrastruktur

Hosting



Motiven für die Suche nach Alternativen

- Beanspruchung von DB Kapazitäten durch AQ
- Behinderung von DB Updates / Upgrades
- Zusatzaufwand hinsichtlich Cloning und Replikation
- Zugangsberechtigung
- Standardisierung Vereinigung

Beanspruchung von DB Kapazitäten durch AQ

Empfehlungen

| SQL Tuning Advisor planen | | | |
|--|----------------------------|------------|---------------|
| Alles auswählen Nichts auswählen Alle Details anzeigen Alle Details ausblenden | | | |
| Auswählen | Details | Kategorie | Vorteil (%) ▾ |
| <input type="checkbox"/> | Ausblenden | SQL Tuning | 12,2 |
| Aktion Optimieren Sie den PL/SQL-Block mit SQL_ID "7hr554xaxwsjc". Hierzu wird auf das Kapitel "PL/SQL-Anwendungen optimieren" "PL/SQL User's Guide and Reference" verwiesen. SQL-Text <code>begin dbms_aqin.aq\$_jms enqueue object message(queue_name => :1, sender_name =>...</code> SQL ID <code>7hr554xaxwsjc</code> | | | |
| <input checked="" type="checkbox"/> | Ausblenden | SQL Tuning | 12,2 |
| Aktion Führen Sie SQL Tuning Advisor für die SQL-Anweisung mit SQL_ID "89cq9kvvh1ffc" aus. Advisor jetzt ausführen Filter SQL-Text <code>insert into "JMSUSER"."GVEINGANG" (q_name, msqid, corrid, priority, state, dela...</code> SQL ID <code>89cq9kvvh1ffc</code> | | | |
| <input checked="" type="checkbox"/> | Anzeigen | SQL Tuning | 4,2 |
| <input checked="" type="checkbox"/> | Anzeigen | SQL Tuning | 3,1 |
| <input checked="" type="checkbox"/> | Anzeigen | SQL Tuning | 0,3 |

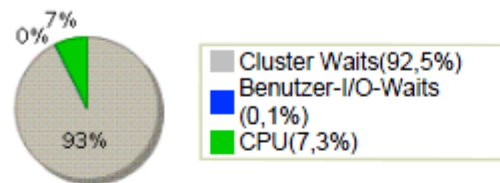
Beanspruchung von DB Kapazitäten durch AQ

```

begin
dbms_aqin.aq$_jms_enqueue_object_mes
sage( queue_name => :1, sender_name
=> :2, sender_addr=> :3,
sender_protocol => :4,
original_msgid => :5, correlation =>
:6, visibility => :7, priority =>
:8, delay
=> :9, expiration=> :10,
relative_msgid => :11,
sequence_deviation => :12,
exception_queue => :13,
payload_type => :14, raw_user_data
=> null, object_message => :15,
recipients => :16, msgid => :17,
gen_payload => :18, signature =>
:19, transformation => :20 ); end;

```

Aktivität nach Waits



Aktivität nach Zeit

Abgelaufene Zeit (s) 14.405,72
 CPU-Zeit (s) 1.061,27
 Wartezeit (s) 13.344,46

Aufgliederung von abgelaufener Zeit

SQL-Zeit (s) 13.999,35
 PL/SQL-Zeit (s) 406,37
 Java-Zeit (s) 0,00

Ausführungsstatistiken

| | Gesamt | Pro Ausführung | Pro Zeile |
|-------------------------|-----------|----------------|-----------|
| Ausführungen | 476.311 | 1 | 1,00 |
| Abgelaufene Zeit (s) | 14.405,72 | 0,03 | 0,03 |
| CPU-Zeit (s) | 1.061,27 | <0.01 | <0.01 |
| Puffer-Lesezugriffe | 3.371.453 | 7,08 | 7,08 |
| Plattenlesezugriffe | 1.583 | <0.01 | <0.01 |
| Direkte Schreibvorgänge | 0 | 0,00 | 0,00 |
| Zeilen | 476.304 | 1,00 | 1 |
| Lesen | 0 | 0,00 | 0,00 |

Andere Statistiken

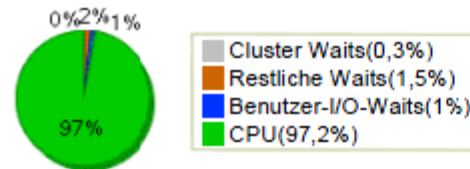
Ausführungen, die alle Zeilen gelesen
 Durchschnittlicher persistenter Speicher
 Durchschnittlicher Laufzeitprozent
 Serialisiert

Untergeordnete Latenz

Beanspruchung von DB Kapazitäten durch AQ

```
select t.schema, t.name, t.flags, q.name,
t.timezone
from system.aq$queue_tables t,
system.aq$queues q
where t.objno = :1 and q.table_objno =
t.objno and q.usage = 0 and NOT ( t.name
in ('DEF$_AQCALL' , 'DEF$_AQERROR') and
t.schema = 'SYSTEM')
```

Aktivität nach Waits



Aktivität nach Zeit

Abgelaufene Zeit (s) 127,59
 CPU-Zeit (s) 124,07
 Wartezeit (s) 3,52

Aufgliederung von abgelaufener

SQL-Zeit (s) 127,59
 PL/SQL-Zeit (s) 0,00
 Java-Zeit (s) 0,00

Ausführungsstatistiken

| | Gesamt | Pro Ausführung | Pro Zeile |
|-------------------------|-----------|----------------|-----------|
| Ausführungen | 1.026.432 | 1 | 1,00 |
| Abgelaufene Zeit (s) | 127,59 | <0.01 | <0.01 |
| CPU-Zeit (s) | 124,07 | <0.01 | <0.01 |
| Puffer-Lesezugriffe | 5.132.154 | 5,00 | 5,00 |
| Plattenlesezugriffe | 22 | <0.01 | <0.01 |
| Direkte Schreibvorgänge | 0 | 0,00 | 0,00 |
| Zeilen | 1.026.430 | 1,00 | 1 |
| Lesen | 1.026.432 | 1,00 | 1,00 |

Andere Statistiken











Ausführungen, die alle Zeilen c
 Durchschnittlicher persisten
 Durchschnittlicher Lau
 Ser

Untergeordn

Beanspruchung von DB Kapazitäten durch AQ

Top Sessions im Status Aktiv

Anzeigen Top Sessions

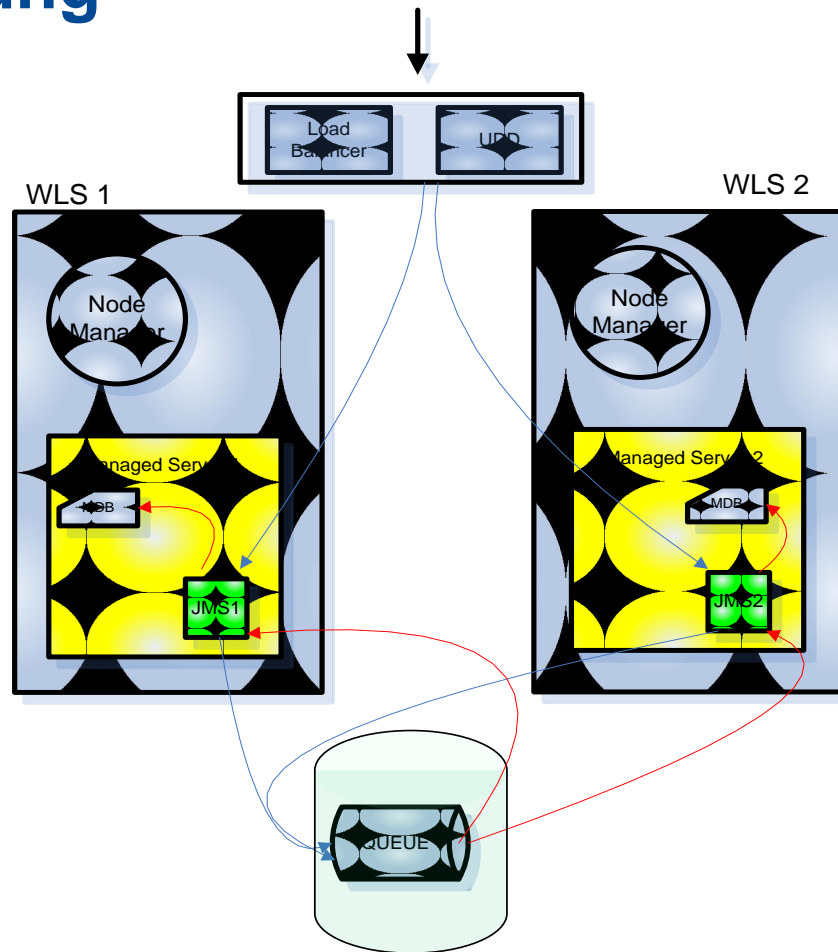
| Aktivität (%) ▾ | Session-ID | Benutzername | Programm |
|--|---------------------|---------------------------|-----------------------------|
|  21,74 | 95 | LORDI_JMS | java@s-app03 (TNS V1-V3) |
|  8,70 | 108 | LORDI_JMS | java@s-app04 (TNS V1-V3) |
|  8,70 | 105 | LORDI_JMS | java@s-app04 (TNS V1-V3) |
|  8,70 | 106 | LORDI_JMS | java@s-app04 (TNS V1-V3) |
|  8,70 | 107 | E..._JMS | java@s-app04 (TNS V1-V3) |
|  8,70 | 109 | E..._JMS | java@s-app04 (TNS V1-V3) |
|  6,52 | 122 | E..._JMS | java@s-app03 (TNS V1-V3) |
|  4,35 | 84 | ..._JMS | java@s-app03 (TNS V1-V3) |
|  4,35 | 127 | ..._JMS | java@s-app03 (TNS V1-V3) |
|  2,17 | 128 | ..._JMS | java@s-app03 (TNS V1-V3) |

Gesamtanzahl von Samples: 46

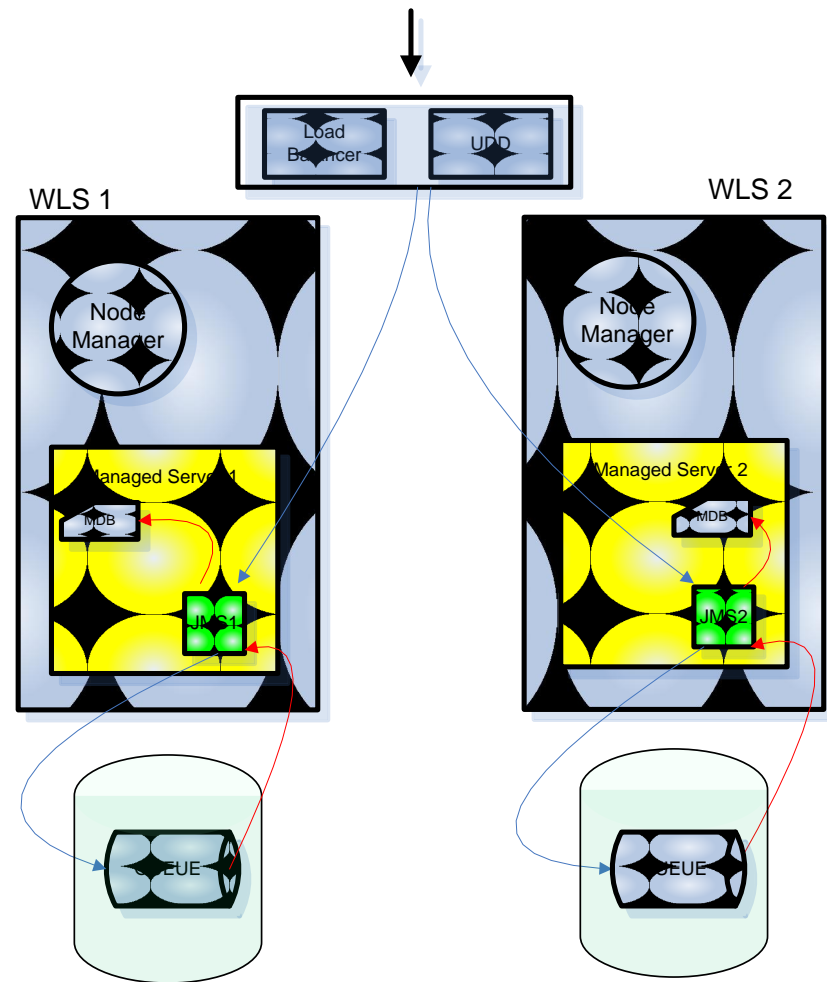
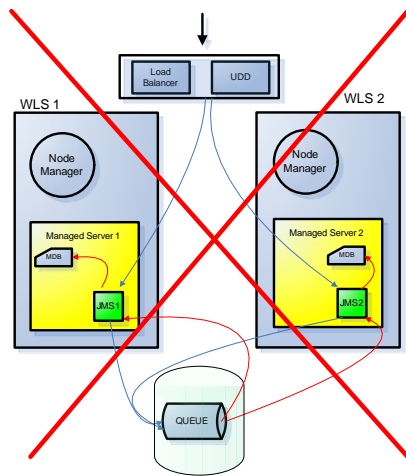
Konzepte und Misskonzepte über das WebLogic JMS

- Clustering im Vergleich mit AQ
- Hochverfügbarkeit / Ausfallsicherheit
- Persistenz

Clustering - Erwartung



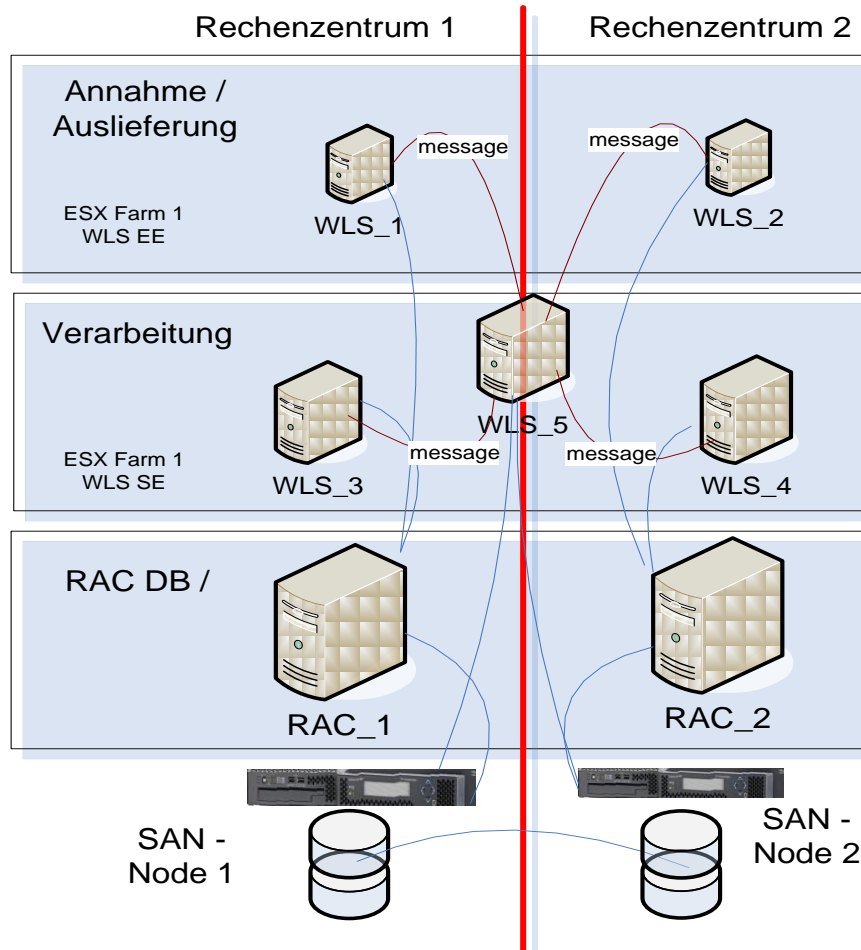
Clustering - Tatsächlich



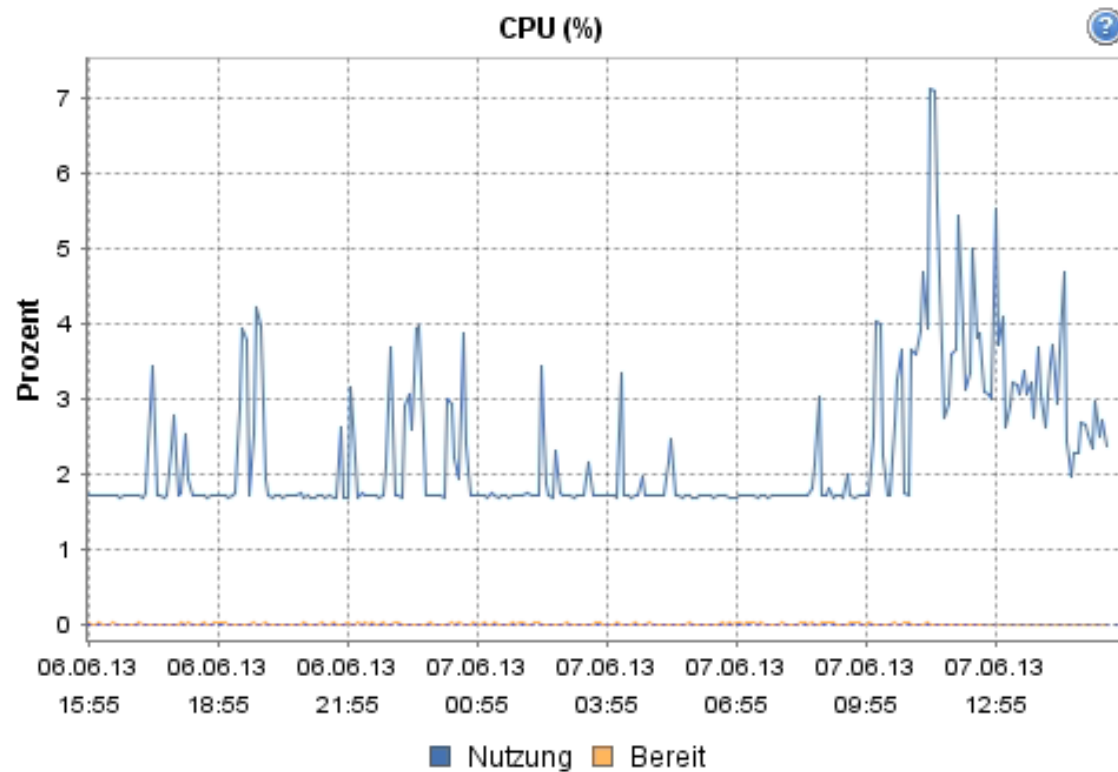
Auf der Suche nach eine Ausfallsichere Lösung

- WebLogic Whole Server Migration
 - Shared Storage ?
 - Erfahrung ?
 - Kosten ?
 - Relocation ohne Unterbrechung ?
- JMS als dedicated WLS Server on VMWare
 - vMotion - Live Migration ohne Unterbrechung
 - DRS – Automatischer Restart nach Hardware/Software Ausfall
 - Rolling Update / Patching ?

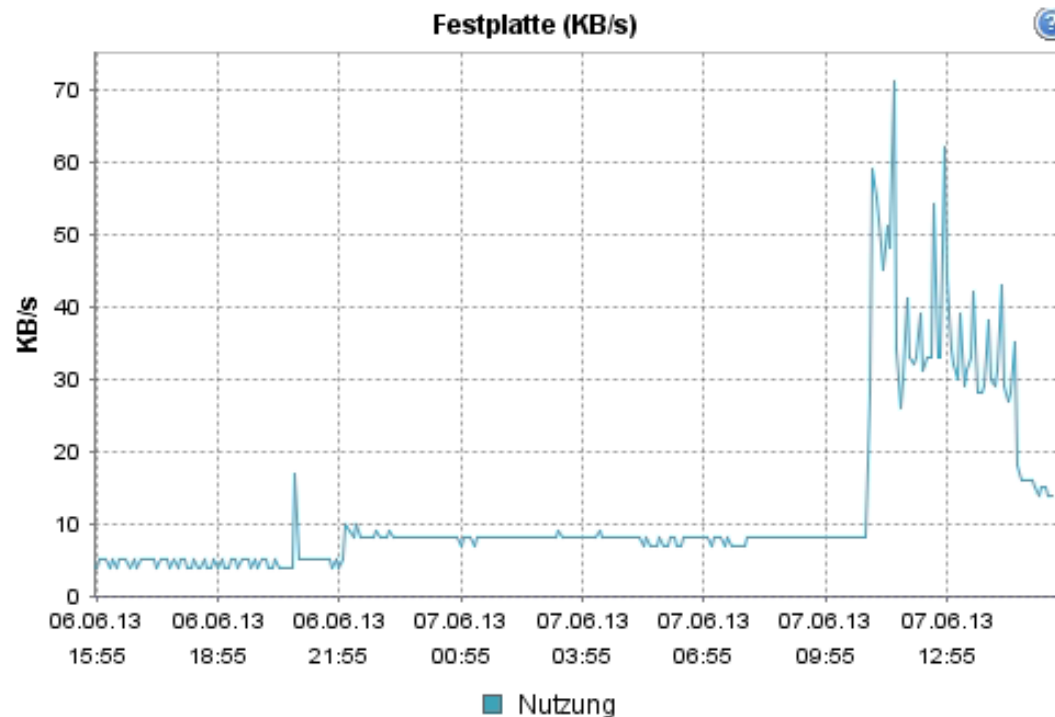
JMS als dedicated WLS Server on VMWare



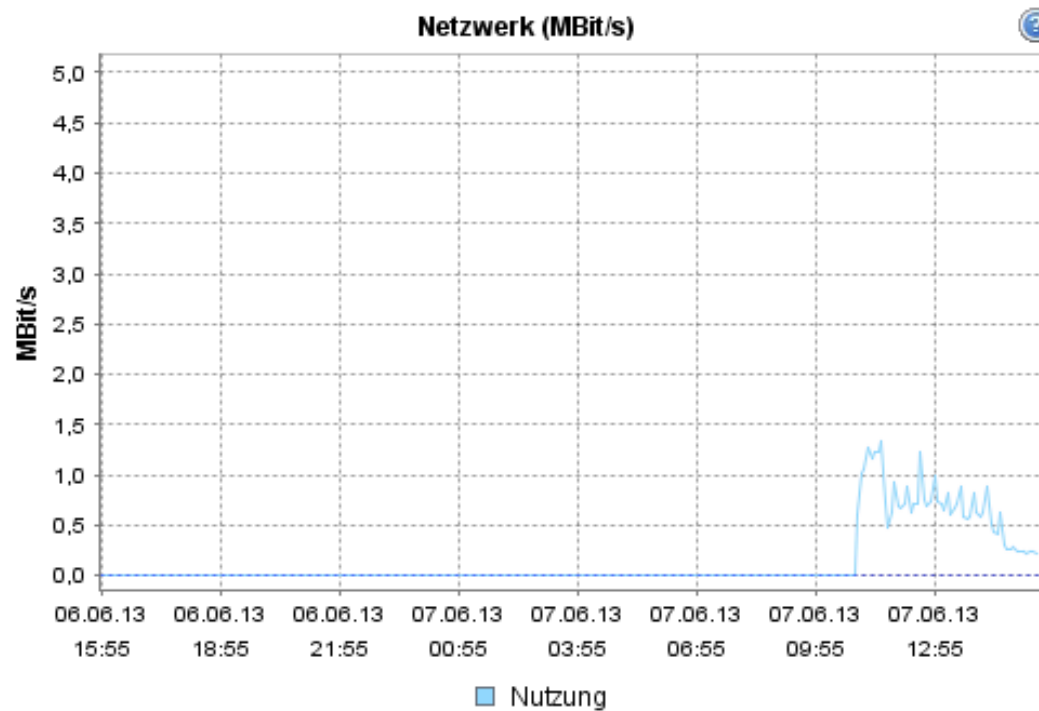
VMWare – Performance Test



VMWare – Performance Test



VMWare – Performance Test



Lizenzkosten-Überlegungen

WebLogic EE

1 Prozessor - € 19.733,00

Server mit 1x4Core Prozessor = € 39.466,00

Pro Kern - € 9.866,50

WebLogic SE

1 Prozessor - € 7.893,00

Server mit 2x6Core Prozessor = € 15.786,00

Pro Kern - € 1.315,50

Lizenzkosten - FEATURE MATRIX

| Benefits | Description | Features | WebLogic Server Standard Edition | WebLogic Server Enterprise Edition | WebLogic Suite |
|---------------------------|--|--|----------------------------------|------------------------------------|----------------|
| Core Messaging | JMS messaging | Point-to-point, publish and subscribe, durable subscribers, XA compliant, XML messaging, unit of order, unit of work | • | • | • |
| | | Asynchronous data processing with message-driven beans | • | • | • |
| | | Oracle WebLogic JMS .NET client, a native .Net runtime library for C# client application access to JMS | • | • | • |
| | | Plug and play with third-party messaging providers, including MQSeries | • | • | • |
| Enterprise Grid Messaging | High availability and scalability message infrastructure | JMS functionality distributed across clusters | | • | • |
| | | Highly-available connection factories and destinations | | • | • |
| | | Automatic migration of messaging/JMS | | • | • |
| | | Store and Forward Messaging | • | • | • |
| | | Integration with Oracle Advanced Queuing (AQ) | • | • | • |

Lizenzkosten - Cluster Support Clause ?

“1.2.2.3.2 Cluster Support

Cluster Support is provided in all WebLogic Server Enterprise Edition and WebLogic Suite Edition licenses, and includes the following capabilities:

...

- Java Messaging Service (JMS) destinations. Note:
 - - A connection factory and a destination can be **targeted to different WebLogic Server instances**.
 - - **Foreign servers** and distributed destinations may be used across multiple WebLogic Server instances.”

Zusammenfassung

Die VM Ware Option ist die bevorzugte Lösung für uns

- Installationsaufwand
- Betriebsaufwand
- Relocation ohne Unterbrechung möglich

Einige Lizenzierungsfragen sind noch offen

Fragen, Ideen, Vorschläge

Q&A

Rossen.Rachev@dguv.de

Thank you for your attention!