



Batch-Frameworks in Java: Architektur und Implementierungen

Stuttgart, 17. Juli 2014, Martin Burkhardt

Vorstellung



Martin Burkhardt, Dipl.-Ing. (FH)
Technische Informatik (FHTE)

Seit 1993 im IT Bereich, seit 1997
Schwerpunkt auf Java-Projekte

Bei Capgemini als Technischer
Architekt und Projektleiter tätig



Consulting, Technologie,
Outsourcing

> 130.000 MA (davon ca. 50.000
Indien), in 2013 > 10 Mrd. Umsatz

Lokale Niederlassung: Stuttgart-
Degerloch (ca. 200 MA)

Info-Stand hier am JFS-2014

Agenda

Batch

Anforderungen

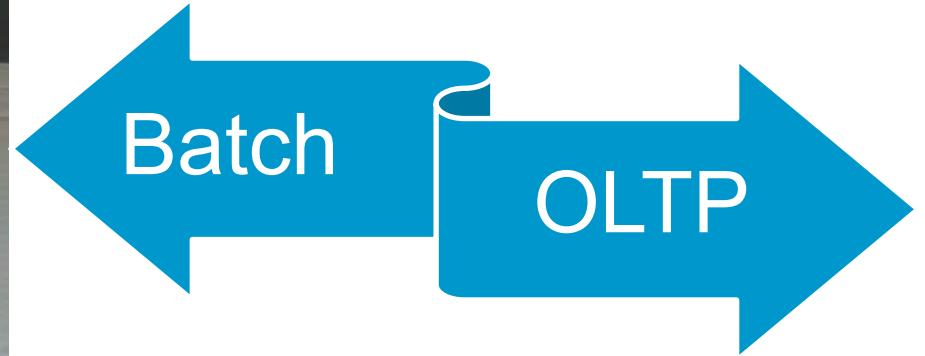
DEMO?

Architekturen /
Implementierungen

Zusammenfassung

Was ist ein Batch?

Batch of Cards (Quelle: Wikipedia / Lochkarte)



Batch-Framework

- Stellt die technische Ablaufumgebung für Batches bereit
- Spezifiziert, wie Batches beschrieben und implementiert werden

Job

- Beschreibt die auszuführende Tätigkeit
- Kann mit Parametern belegt werden (Job + Parameter = Job-Instanz)
- Kann gestartet werden (laufende Job-Instanz = Job-Execution)

Step

- Ein Job besteht aus einzelnen Schritten (Steps)
- Typisch für die Aufgaben in einen Step:
 - werden von einer Transaktion umschlossen
 - können u.U. parallel laufen

Typische Anforderungen an ein Batch-Framework

Job Definition

- Ausschlüsse..(während A läuft darf B nicht)
- Prioritäten
- Nebenläufigkeit (von Jobs und Steps)
- Implementierung (typischerweise ein Verweise auf Programmcode)

Job Ausführung

- Start
 - explizit
 - ereignisgesteuert
 - zeitgesteuerter
- Serien von Jobs
- Verteilte Ausführung
- Lastverteilung

Job Steuerung

- Benutzerschnittstelle für Start/Stop/Abbruch von Jobs
- Job Ergebnisse liefern/protollieren, Job Verlauf dokumentieren (Logging)
- Ressourcen Überwachung (z.B. keine weiteren Jobs bei Speicherknappheit)

Fehlerbehandlung..Abort..Skip..Retry..Transaktionen..Checkpoints

JSR-352 Übersicht









■ Historie

- JSR-352 wurde im Mai 2013 finalisiert
- Im JEE7 enthalten (javax.batch.*)
- Referenzimplementierung JBatch kann auch in der Java SE verwendet werden..
- Übernimmt Konzepte aus IBM Produkten sowie aus Spring Batch

■ Features

- Jobs in XML definieren
- Item processing (auch parallel)
- Kein Scheduling Framework

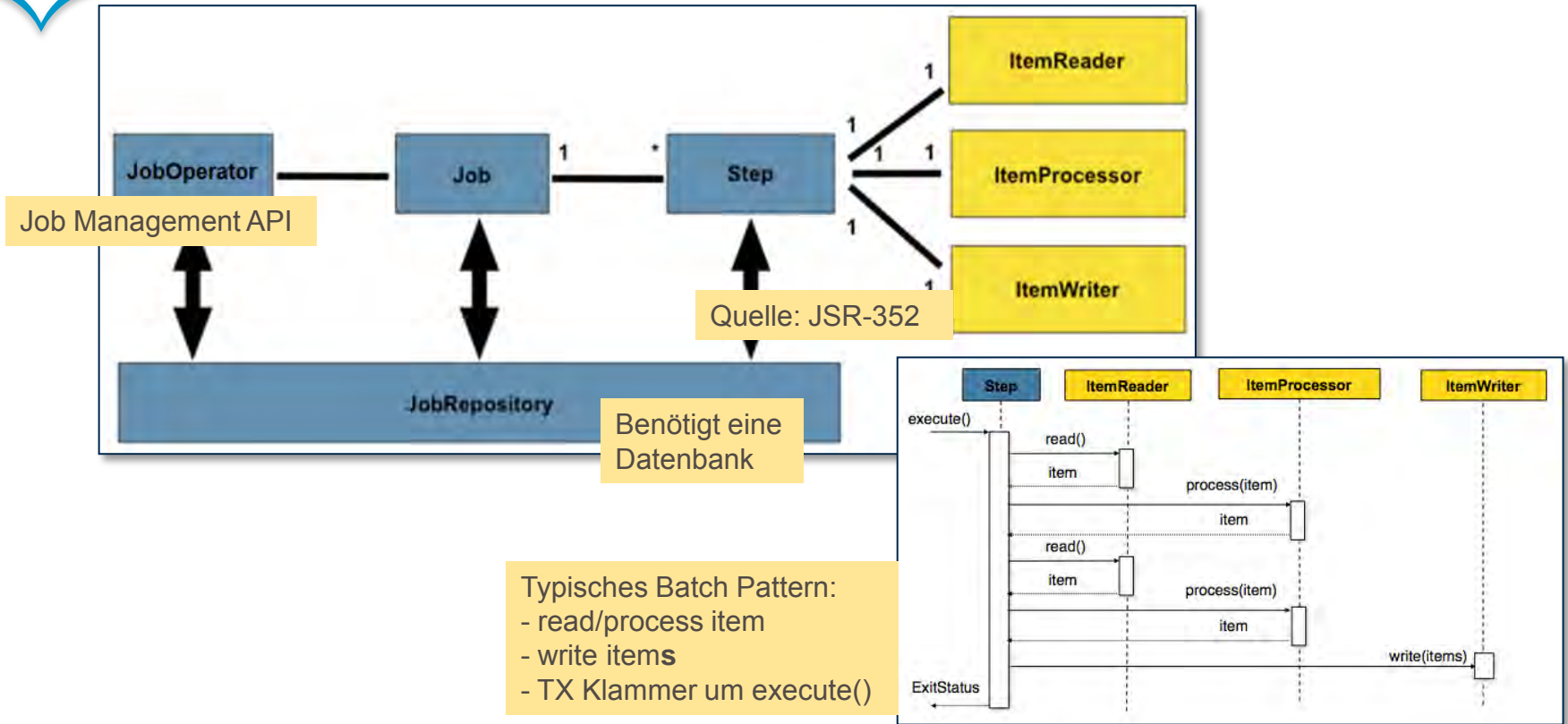
JBatch Archiv

 derby.jar
 javax.batch.api.jar
 javax.inject.jar
 jsr352-RI-spi.jar
 jsr352-SE-RI-javadoc.jar
 jsr352-SE-RI-runtime.jar
 LICENSE.txt
 NOTICE.txt

Quelle: JSR-352

7	Domain Language of Batch
7.1	Job
7.1.1	JobInstance
7.1.2	JobParameters
7.1.3	JobExecution
7.2	Step
7.2.1	StepExecution
7.3	JobOperator
7.4	Job Repository.....
7.5	ItemReader
7.6	ItemWriter
7.7	ItemProcessor
7.8	Chunk-oriented Processing
7.9	Batch Checkpoints.....

JSR-352 Architektur



Typisches Batch Pattern:
- read/process item
- write items
- TX Klammer um execute()

JSR-352 Job Specification Language (JSL) 1/2

```
<job id="batch-demo" xmlns="http://xmlns.jcp.org/xml/ns/javaee" version="1.0">
  <properties>
    <property name="input_file" value="C:/Us... />
    <property name="output_file" value="C:/Us... />
  </properties>

  <step id="first-step" next="second-step">
    <listeners>
      <listener ref="com.capgemini.jfs2014.BatchDemoStepListener" />
    </listeners>
    <chunk item-count="5">
      <reader ref="com.capgemini.jfs2014.BatchDemoItemReader" />
      <processor
        ref="com.capgemini.jfs2014.BatchDemoItemProcessor" />
      <writer ref="com.capgemini.jfs2014.BatchDemoItemWriter" />
    </chunk>
  </step>

  <step id="second-step">
    <batchlet ref="com.capgemini.jfs2014.BatchDemoBatchlet" />
  </step>
</job>
```

Die meisten Elemente kennen <properties>

Ein step kann seinen Nachfolger festlegen

Viele Elemente können mit Listnern bestückt werden

Ein step kann chunk **oder** batchlet enthalten

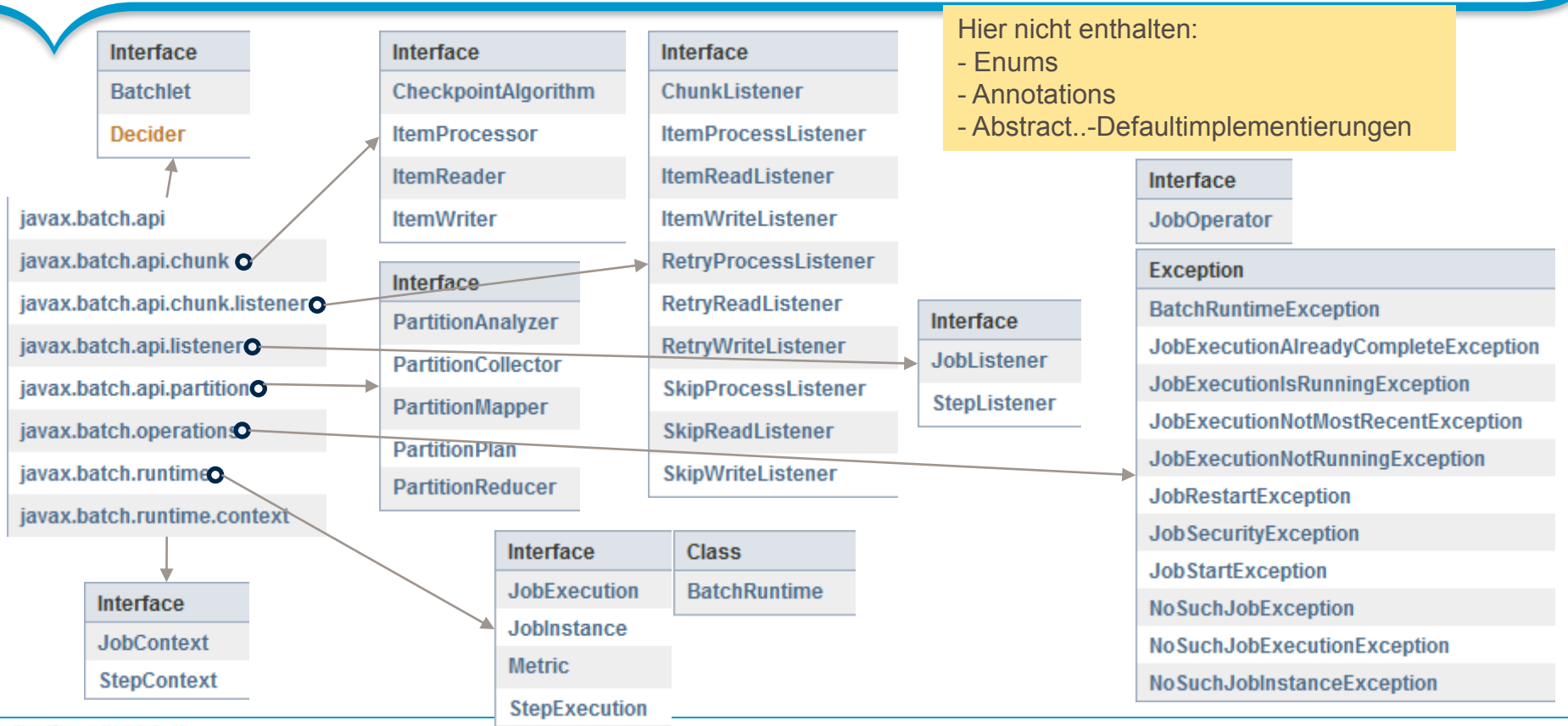
JSR-352 Job Specification Language (JSL) 2/2

- Job*
 - ..Job Level Listeners
 - ..Job Level Exception Handling
- Step*
 - ..Chunk
 -Reader*
 -Processor*
 -Writer*
 -Chunk Exception Handling
 -Skipping Exceptions
 -Retrying Exceptions
 -Retry and Skip the Same Exception
 -Default Retry Behavior - Rollback
 -Preventing Rollback During Retry
 -Checkpoint Algorithm*
 - ..Batchlet*
 -Batchlet Exception Handling
 - ..Step Level Listeners*
 - ..Step Sequence
 - ...Step Partitioning*
 -Partition Plan
 -Partition Mapper*
 -Partition Reducer*
 -Partition Collector*
 -Partition Analyzer*
 - ..Step Exception Handling
 - Flow
 - Split
 - Decision*
 - ..Decision Exception Handling
 - Transition Elements
 - ..Next Element
 - ..Fail Element
 - ..End Element
 - ..Stop Element
 - Batch and Exit Status
 - ..Batch and Exit Status for Steps
 - ..Exit Status for Partitioned Steps

- Job XML Substitution
 - ..Substitution Processing Rules
 -jobParameters Substitution Operator
 -jobProperties Substitution Operator
 -systemProperties Substitution Operator
 -partitionPlan Substitution Operator
 - ...Substitution Expression Default
 -Property Resolution Rule
 -Undefined Target Name Rule
 -Job Restart Rule

* Diese Elemente kennen das Unterelement „Properties“

JSR-352 Programmiermodell



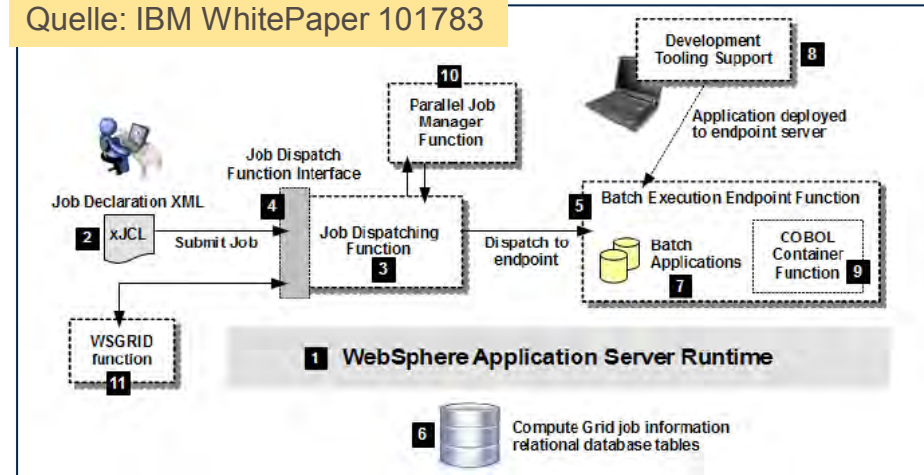
JSR-352 Zusammenfassung

- Version 1.0
- JEE (z.Zt. Glassfish) und JSE (z.Zt. jbatch RI)
- Keine Ressourcen-Verwaltung (z.B. Hauptspeicher Überwachung)
- Kein Monitoring, keine Admin UI
- JSL: spezifiziert den Job, ist aber keine Job-Steuerung
 - Keine zeit- oder eventbasierte Jobasuführung
- Parallelisierung von Aufgaben über *partition* und/oder *split*

WebSphere Batch 1/2

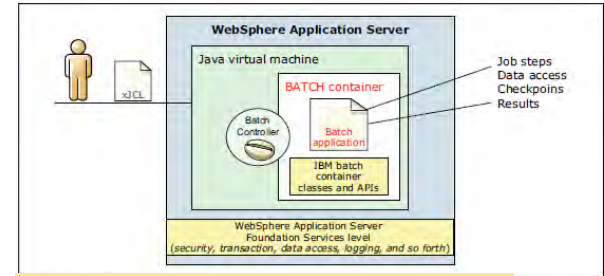
- WAS 8.5 Java Batch = WAS 8 + Compute Grid 8
- WAS unterscheidet
 - Batch applications -> steps, I/O, transactions
 - Compute-intensive applications -> run()
- XML job control language = xJCL (basiert auf z/OS JCL)
- Ein Dispatcher verteilt die Arbeit auf „endpoints“ (typischerweise WAS Knoten)
- WAS Java Batch ist im WAS 8.5 Liberty Profile nicht enthalten

Quelle: IBM WhitePaper 101783

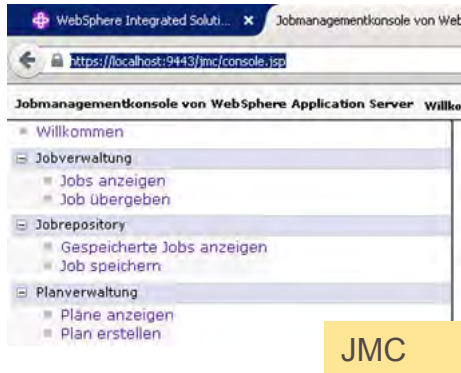


WebSphere Batch 2/2

- Batch Container: ein EJB Container, in dem die Batch-EARs deployed werden
- Ein BatchControllerEJB (2.1!) ist die Brücke zum JobScheduler



Quelle: IBM WhitePaper 101783



- Für Admins (und User): die Job Management Console
- Hier werden u.a. xJCLs hochgeladen
- Batch Data Stream Framework (BDSF): unterstützt verschieden I/O Kanäle:
 - JDBC, JPA, Text- und Binärdateien

WebSphere Batch Zusammenfassung

- Lange Historie, bewährte Konzepte aus der z/OS Welt wurde übernommen
- “Enterprise” Features, z.B. geclustert Batch Ausführung
- Ausführliches Job Log
- Integration mit weiteren Enterprise Produkten (Tivoli, WSGRID) möglich
- WebSphere JSR352 Roadmap noch nicht veröffentlicht

Feature	JSR 352	WAS Batch
Transactional Batch Model	Yes	Yes
Compute Intensive Model (Batchlet)	Yes	Yes
Partitioned Step Execution	Yes	Yes
Job and Step Contexts	Yes	Yes
Job and Step Listeners	Yes	Yes
Job Specification Language (JSL)	Yes	Yes
JSL Substitution	Yes	Yes
Operations Interface	Yes	Yes
Concurrent Steps	Yes	NO
Restart Substitution	Yes	NO
CDI Integration	Yes	NO
Job Dispatcher	NO	Yes
Job Logs	NO	Yes
Distributed Parallelization	NO	Yes

Quelle: IBM Java Batch Solutions

Capgemini Lösung – kurz skizziert 1/2

- Kombination aus WAS Scheduler und Eigenentwicklung
- Monitor UI als Teil der fachlichen Anwendung
- Ressourcen Steuerung
 - Threads
 - Speicherverbrauch
- Job Steuerung
 - Abhängigkeiten
 - Ausschlußkriterien
 - Zeitsteuerung

```
TAPlanCreationBatchConfigurationBO.memoryUsage = 400  
TAPlanCreationBatchConfigurationBO.maxThreads=8
```

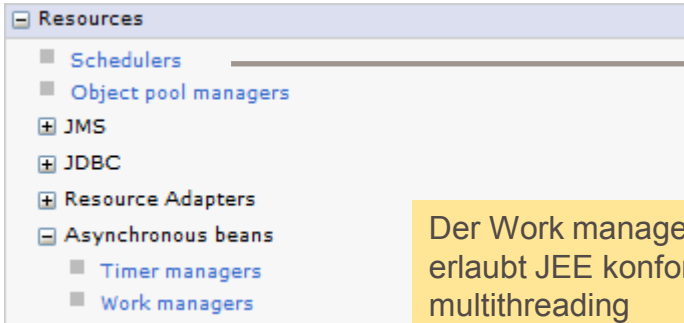
Konfigurationsdateien

RestartableBatchService **Java Code**

```
Ⓜ F MAX_PARAMETER_LENGTH : int  
Ⓜ F MAX_TASK_PARAMETERS : int  
● finalizeTask(Method, TaskState, TaskExecutionResultEnum) : void  
● getAdditionalInformation(String[]) : String  
● getDisplayName(Locale) : String  
● getExcludedBatches() : Class[]  
● getExecutionDescriptor(String[]) : BatchExecutionDescriptor  
● getProjectedMemoryUsage(String[]) : int  
● initializeTask(Method, String[], String[]) : TaskState  
● restartTask(Method, String[], String[], String) : TaskState  
● shouldBeSkipped(String[]) : boolean
```


Capgemini Lösung – kurz skizziert 2/2

- Vor WebSphere 8.5: Scheduler und Work manager



Der Work manager erlaubt JEE konformes multithreading

NEXTFIRETIME	STARTBYINTERVAL	STARTBYTIME	VALIDFROMTIME	VALIDTOTIME	REPEATINTERVAL
1396362510971	<null>	<null>	1396362510971	<null>	<null>
1402495140230	<null>	<null>	1402314540783	<null>	0 9/10 *** ?
1402495020233	<null>	<null>	1402314420695	<null>	0 7/10 *** ?
1402494900529	<null>	<null>	1402314300778	<null>	0 5/10 *** ?
1402495440969	<null>	<null>	1402315440087	<null>	0 4/30 *** ?
1402494660212	<null>	<null>	1402314660770	<null>	0 1/10 *** ?
1402527600087	<null>	<null>	1402354800061	<null>	0 00 1 *** ?
1402527600115	<null>	<null>	1402354800069	<null>	0 00 1 *** ?
1402527600120	<null>	<null>	1402354800076	<null>	0 00 1 *** ?
1402527600089	<null>	<null>	1402354800053	<null>	0 00 1 *** ?
1402506000656	<null>	<null>	1402336200887	<null>	0 0/10 6-7 19-20 *** ?

WAS Scheduler benötigt eine DB für die Job-Verwaltung

Abschlussbetrachtung

- JSR-352 ist auf einem guten Weg
- JEE Implementierungen derzeit noch rar (evtl. in in WAS 9?)
- Wer heute ein Lösung auf WebSphere-Basis benötigt kann WAS Java Batch verwenden
 - Ein Migrationspfad von WAS Java Batch auf JSR-352 scheint wahrscheinlich
 - Bei intensiver Nutzung der WAS Features (z.B. BDS) kann eine Migration aufwendig werden
- Es gibt noch mehr:
 - SpringBatch (ist mit Version 3 JSR-352 konform)
 - JEE7..App Server z.Zt. noch rar: WildFly 8.0, Glasfish 4.0, TMAX JEUS 8

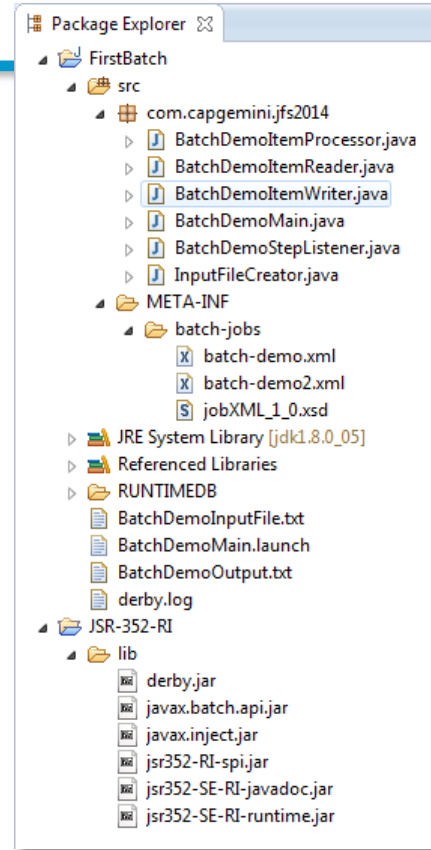
Batch Demo – JSR-352

Auf Basis der jbatch RI

Zeilen eines Textfiles in 5-er
Chunks verarbeiten

Einfache JSL

Listener Konzept



Referenzen

JSR-352

- <https://jcp.org/en/jsr/detail?id=352>
- <http://docs.oracle.com/javaee/7/api/>
- <https://java.net/projects/jbatch>

Spring Batch

- <http://docs.spring.io/spring-batch/trunk/reference/html/index.html>

JEE8 Tutorial (Kapitel Batch)

- <http://docs.oracle.com/javaee/7/tutorial/doc/batch-processing.htm>

IBM WebSphere Java Batch

- White Paper 101783
- WAS InfoCenter 8.5
<http://pic.dhe.ibm.com/infocenter/wasinfo/v8r5/index.jsp>

Capgemini

- www.capgemini.de
- Martin.Burkhardt@capgemini.com
- Stand hier am JFS-2014

> END OF PPT BATCH..

-> Fragen??



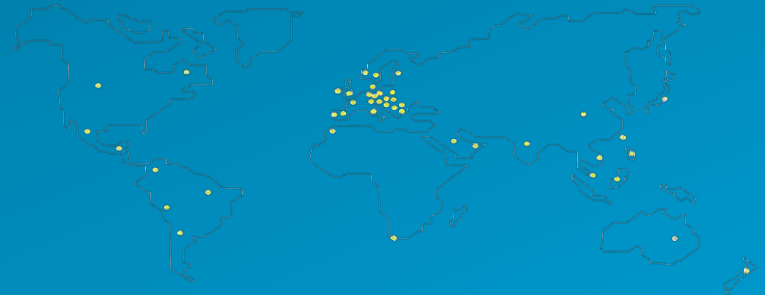
People matter, results count.

About Capgemini



With more than 130,000 people in over 40 countries, Capgemini is one of the world's foremost providers of consulting, technology and outsourcing services. The Group reported 2013 global revenues of .EUR 10.1 billion.

Together with its clients, Capgemini creates and delivers business and technology solutions that fit their needs and drive the results they want. A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.



www.capgemini.com

