

IBM Education Assistance for z/OS V2R1

Item: IBM Batch Programming Model

Element/Component: BCP Batch Runtime



Agenda

- Trademarks
- Presentation Objectives
- Overview
- Usage & Invocation
- Interactions & Dependencies
- Installation
- Appendix



Trademarks

- See url <http://www.ibm.com/legal/copytrade.shtml> for a list of trademarks.



Presentation Objectives

- Purpose
- How to install
- How to invoke
- How to find information in the publications



Overview

- Problem Statement / Need Addressed
 - Applications need to be reused in multiple environments
- Solution
 - Re-engineer application using IBM Batch Programming Model
- Benefit / Value
 - Applications can be deployed on WAS distributed, WAS z/OS, or if compliant with limitations of Batch Runtime, a z/OS batch job



Usage & Invocation

- Batch Runtime is invoked through JCL
 - Sample JCL procedure BCDPROC
 - Sample invocation JCL BCDBATCH

- Customize BCDBATCH job
 - Specify application language (XJCL)
 - Set CLASSPATH and LIBPATH needed by application
 - Provide xJCL to describe batch job

- Batch Runtime support is based on the Websphere Compute Grid batch programming model
 - Not all functions are supported under z/OS: see documentation



Usage & Invocation

- IBM Batch Programming Model
 - Java-centric means of batch programming
 - Policy driven checkpoint processing
 - Global transaction based
 - Container driven restart processing

- Job defined as a series of steps
 - Files required by the step are defined by batch data streams
 - File system, MVS data sets, DB2 tables
 - Each step runs sequentially and is transactionally independent from each other

- Job is described by xJCL



Usage & Invocation

- xJCL is an XML based file which describes the batch job
 - xJCL must be provided by the user when submitting the job
 - Use Rational Application Developer (RAD) xJCL editor to create
 - Or code by hand

- Each batch step contains business logic to run a portion of the job
 - Typically reads a record, performs business logic, and gets next record
 - Must conform to `com.ibm.websphere.batch.BatchJobStepInterface`
 - Implement methods `createJobStep`, `processJobStep`, `destroyJobStep`, and `setProperties`



Usage & Invocation

- New Batch Runtime JCL statements
 - //BCDXJCL names a file containing the xJCL
 - Usually resides in file system
 - //BCDXPROP names a file containing substitution properties to be applied to xJCL
 - Usually is provided inline within the job



Usage & Invocation

- New Batch Runtime control statements (//BCDIN)
 - `bcd.applicationLanguage=XJCL`
 - Specifies that xJCL is being used
 - `bcd.xJCLEncoding=encoding-name`
 - Names encoding to be used when reading xJCL file
 - Useful when default file encoding for the JVM is different than the encoding of the xJCL
 - JVM might have default encoding of ISO8859-1
 - xJCL might have encoding of IBM-1047



Usage & Invocation

- New Batch Runtime control statements (//BCDIN)
 - `bcd.xJCLRestartEnabled=true | false`
 - Specifies whether job is restartable
 - Causes state information to be saved in file system
 - `bcd.xJCLRestartHome=path`
 - Names home directory where restart files will be placed
 - Default is working directory for user
 - `bcd.xJCLRestartJobId=jobid`
 - Indicates this is a restart run and the jobid to be restarted



Usage & Invocation Sample JCL

```
//BATCH EXEC BCDPROC,REGION=0M,LOGLVL='+T',VERSION=61
//*
//BCDXJCL DD PATH='yourpathxJCL.xml',
//      PATHOPTS=ORDONLY
//*
//BCDXPROP DD *
inputDataStream=inputFile.txt
outputDataStream=outputFile.txt
fileEncoding=IBM-1047
debugEnabled=true
//BCDIN DD *
bcd.applicationLanguage=XJCL
bcd.xJCLEncoding=ibm-1047
bcd.xJCLRestartEnabled=true
#
bcd.supportClass.1=com.ibm.db2.jcc.t2zos.T2zosBatchContainerSupport
```



Usage & Invocation Sample xJCL

```
<?xml version="1.0" encoding="ASCII"?>
<job xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
     xsi:noNamespaceSchemaLocation="http://core.wcg.etoools.ibm.com/xsd/xJCL.xsd"
     name="test" default-application-name="test">
  <job-type>Batch</job-type>
  <jndi-name>ejb/XDCGIVTBatchController</jndi-name>
  <step-scheduling-criteria>
    <scheduling-mode>sequential</scheduling-mode>
  </step-scheduling-criteria>
  <checkpoint-algorithm name="Checkpoint">
    <classname>com.ibm.wsspi.batch.checkpointalgorithms.recordbased</classname>
    <props>
      <prop name="TransactionTimeOut" value="1" />
      <prop name="recordcount" value="10" />
    </props>
  </checkpoint-algorithm>
  <substitution-props>
    <prop name="inputStream" value="inputFile.txt" />
    <prop name="outputStream" value="outputFile.txt" />
  </substitution-props>
```



Usage & Invocation Sample xJCL

```
<job-step name="Step1">
  <classname>com.ibm.zos.batch.container.test.ibmpm.TestRestartBatchStep</classname>
  <checkpoint-algorithm-ref name="Checkpoint" />
  <batch-data-streams>
    <bds>
      <logical-name>inputStream</logical-name>
      <impl-class>com.ibm.websphere.batch.devframework.datastreams.patterns.
TextFileReader</impl-class>
      <props>
        <prop name="PATTERN_IMPL_CLASS"
          value="com.ibm.websphere.batch.samples.tests.bds.EchoReader" />
        <prop name="FILENAME" value="{inputStream}" />
        <prop name="debug" value="true" />
      </props>
    </bds>
    ...
  </batch-data-streams>
</job-step>
</job>
```



Interactions & Dependencies

- Software Dependencies
 - IBM 31-bit or 64-bit SDK for Java Technology Edition, V6.0.1 or higher
 - jZOS launcher required (distributed with SDK)
- Hardware Dependencies
 - None
- Exploiters
 - None



Installation

- Batch Runtime is implemented in Java
 - Installs into existing path /usr/lpp/bcp

- BCDPROC is installed into SYS1.PROCLIB
 - Check for modifications needed

- Sample JCL BCDBATCH is installed in SYS1.SAMPLIB
 - Copy to your private JCL data set and customize as needed



Appendix

- Documentation
 - z/OS Batch Runtime: Planning and User's Guide (SA23-7270)
 - Websphere Info Center

