
 <p>26. DECUS Symposium</p>	<h2>Java auf OpenVMS – Ready for Prime Time</h2> <p>Hans Bachner Software Consultant Hewlett-Packard 9. April 2003 2105</p>
--------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------

<h2>Was ist Java ?</h2> <ul style="list-style-type: none">• Programmiersprache / Quellcode• Zwischensprache / Bytecode• Virtual Machine• Entwicklungs- und Laufzeitumgebung • Java Software Development Kit (SDK/JDK)• lizenziert von Sun Microsystems™• keine Anwendung, sondern eine Umgebung mit Werkzeugen • „Write Once, Run Everywhere.“™ (Sun).• „Write Once, Debug Everywhere.“ (unknown). <p>9. April 2003 DECUS 2003 - Java auf OpenVMS page 2</p>	
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------

was bisher geschah



- Lizenz von Sun – 1996
 - Java SDK/RTE 1.1.6
 - Java SDK/RTE 1.1.8
 - J2SDK 1.2.2
 - J2SDK 1.3.0
 - J2SDK 1.3.1
 - J2SDK 1.4.0
 - **J2SDK 1.4.1 beta**
-
- RTE – Saveset, zum Einbinden in eigene Anwendung

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 4

Eigenschaften



- POSIX Threads Implementierung
 - sehr performant in Multiprozessor-Umgebung
- UNIX-ähnliche Befehlszeilen-Optionen
 - DCL-ähnliche Qualifier nur in den ersten Versionen
- die Java-Welt erwartet Unix-/Windows Filespezifikationen
 - verschiedene Optionen für Abbildung auf ODS-2/ODS-5 vorhanden

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 5

Installationsvoraussetzungen



- OpenVMS V7.2-2
 - VMS722_ACRTL-V0200
 - VMS722_UPDATE-V0100
 - VMS722_RMS-V0200 (aktuell: V0400)
- OpenVMS V7.3
 - VMS73_LIBRTL-V0200
 - VMS73_UPDATE-V0100
 - VMS73_SYS-V0400 (aktuell: V0500)
 - VMS73_ACRTL-V0300
 - VMS73_PTHREAD-V0200 (aktuell: V0300)
- OpenVMS V7.3-1
 - VMS731_ACRTL-V0100

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 6

Installationsverzeichnisse



- es können problemlos mehrere Versionen des SDK installiert und auch parallel verwendet werden
 - V1.1.x — SYS\$COMMON:[JAVA]
 - V1.2.2 — SYS\$COMMON:[JAVA\$122]
 - V1.3.0 — SYS\$COMMON:[JAVA\$130]
 - V1.3.1 — SYS\$COMMON:[JAVA\$131]
 - V1.4.0 — SYS\$COMMON:[JAVA\$140]
 - ...

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 7

Verwendung von Java

- Umgebung einrichten


```
$ @SYS$COMMON:[JAVA$140.COM] JAVA$140_SETUP
$ @SYS$COMMON:[JAVA$140.COM] JAVA$140_SETUP FAST

$ @SYS$MANAGER:JAVA$SETUP          ! V1.1.8
```
- Umschalten zwischen Versionen

z.B.

```
$ @SYS$COMMON:[JAVA$140.COM] JAVA$140_CANCEL_SETUP
$ @SYS$COMMON:[JAVA$131.COM] JAVA$131_SETUP
```

9. April 2003 DECUS 2003 - Java auf OpenVMS page 8

Groß-/Kleinschreibung

- Java ist **sehr** „case sensitive“


```
$ type myworld.java
class Myworld {
    public static void main (String args[ ]) {
        System.out.println("Hello world");
    }
}
$ javac Myworld.java
$ java myworld.class
Can't find class myworld.class
$ java Myworld
Can't find class myworld
$ java "Myworld"
Hello world
```

9. April 2003 DECUS 2003 - Java auf OpenVMS page 9

Groß-/Kleinschreibung



- Abhilfe durch ECO für C-Runtime-Library

```
$ DEFINE /JOB DECC$ARGV_PARSE_STYLE ENABLE
$ DEFINE /JOB DECC$EFS_CASE_PRESERVE ENABLE
$ DEFINE /JOB DECC$EFS_CASE_SPECIAL ENABLE
$ SET PROCESS /PARSE=EXTENDED
$ java Myworld
Hello world
```

- automatische Bestimmung der Schreibweise für *javac* und *jar* abschalten, wenn nicht benötigt

```
$ DEFINE /JOB JAVA$OMIT_CASE_CHECK TRUE
```

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 10

Filespezifikationen



- Java erwartet Filespezifikationen im Unix-Stil
- logischer Name JAVA\$FILENAME_CONTROLS gibt vor, wie diese in VMS-kompatible Pfad- und Dateinamen umgesetzt werden
- Standard:


```
$ define JAVA$FILENAME_CONTROLS -1
```
- Performancegewinn durch Optimierung
 - die meisten Optionen erlauben Alternativen, von denen eine auszuwählen ist, z.B.
 - „Support Multi dot in file, keeping the last dot“
 - „Support Multi dot in file, keeping the first dot“
 - Bitmaske als Wert für JAVA\$FILENAME_CONTROLS

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 11

Classpath



- Unix Style

```
$ define classpath -
  "/mydisk/myapp/classes.jar:/sys$common/java$131/lib/jdk$131_classes.zip:"
- maximal 255 Zeichen!
```

- OpenVMS Style

```
$ define java$classpath -
  mydisk:[myapp]classes.jar,
  sys$common:[java$131.lib]jdk$131_classes.zip,
  []
```

DCL Zeilenlänge



- limitiert auf 255 Zeichen (1024 mit Fortsetzungszeilen)

- Umgehungsmöglichkeiten:

- `$ javac -v filespec`

- `$ define JAVA$ENABLE_ENVIRONMENT_EXPANSION TRUE`
 - erlaubt Referenz auf Symbole und logische Namen
`$ define my_classpath [],mydisk:[myapp]lib.jar`
`$ my_options = "-Xgc:compacting -XO"`
`$ java -cp $my_classpath $my_options class`

Wie geht's weiter?



- Java 1.4.0-1 — jetzt verfügbar
- Java 1.3.1-6 — Juni 2003
 - Bugfixes aus Sun's 1.3.1_08
- Java 1.4.1 — beta verfügbar, final (geplant) Juni 2003
- Java 1.4.2 — (geplant) Ende 2003
- Bugfix-Versionen abhängig von Sun Releases

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 1.4

Wie geht's weiter?



```

%EXECINIT-I-PERCPU, initializing per-CPU database
%EXECINIT-I-INITS, calling initialization routines
Ring buffer start,length = G20944F8,8000 Ring Buffer Next Ptr= 820af350

OpenVMS (TM) Itanium Operating System, Version X9LS-02L

Brk 0 at 80010310

80010310!          break.m          0800003 █
  
```

- Itanium Unterstützung geplant für OpenVMS V8.1

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 1.5

CSWS_JAVA (Tomcat)



- derzeit verfügbar: CSWS_JAVA V2.0
- basiert auf Tomcat 4.0.4

- enthält:
 - Tomcat / Catalina
 - Ant (teilweise)
 - JSPC
 - mod_jk zur Anbindung an CSWS

- implementiert
 - Java Server Pages 1.2
 - Java Servlets 2.3

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 16

Installationsvoraussetzungen



- OpenVMS V7.2-2
- alle für Java notwendigen ECOs
- VMS7nn_PTHREAD
- VMS/nn_ACRTL

- ODS-5 Platte erforderlich

- nicht zwingend im CSWS-Verzeichnis

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 17

Apache JServ



- JServ war als CSWS_JSERV erhältlich
- wurde in CSWS_JAVA V1.0 integriert
 - parallele Verwendung beider Technologien möglich
- ist in CSWS_JAVA V2.0 nicht mehr enthalten
 - weiterhin V1.1 verwenden

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 18

Konfiguration



```
$ @SYS$MANAGER:APACHE$JAKARTA
```

- Einstellung des gewünschten Benutzernamens
- Setzen der Zugriffsrechte auf Verzeichnisbaum (ACLs)
- Integration mit Apache (mod_jk)
- Tomcat kann auch als Standalone-Server betrieben werden (ohne Apache)
 - Port 8080

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 19

Konfiguration



- .tomcatrc
 - im Login-Verzeichnis
 - Auswahl der Java-Version
 - automatisch: V1.4.1, V1.4.0, V1.3.1
 - manuell:

```
$ DEFINE APACHE$JAKARTA_JDK_SETUP -  
    mydisk:[mydir]Setup_jakarta_env.com  
$ DEFINE APACHE$JAKARTA_USER_CLASSPATH -  
    SYS$COMMON:[JAVA$1nn.LIB]TOOLS.JAR
```
 - Auswahl der Java Virtual Machine
 - wenn eine der oben genannten Java-Versionen verwendet wird:

```
$ DEFINE APACHE$JAKARTA_USE_FASTVM TRUE
```

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 20

for your reference...



- www.hp.com/go/openvms
- www.hp.com/go/java

- Release Notes / Installation Guide, Kit zum Download, sonstige Info:
 - www.compaq.com/java/alpha
 - www.openvms.compaq.com/openvms/products/ips/apache

9. April 2003

DECUS 2003 - Java auf OpenVMS

page 21

Fragen?



9. April 2003 DECUS 2003 - Java auf OpenVMS page 22

