

MyCoRe Installation mit IBM CM8.2

- Kurzanleitung -

Kathleen Krebs

7. April 2005

Inhaltsverzeichnis

1. Voraussetzungen für eine MyCoRe Anwendung.....	3
2. Download und Installation des MyCoRe Kerns.....	3
3. Die MyCoRe Beispielanwendung.....	4
4. Installation der Webanwendung.....	7

Vorwort

Dieses Dokument beinhaltet eine Kurzanleitung zur Installation von MyCoRe mit IBM CM8.2. Ein bereits installiertes CM-System 8.2, DB2 8.2, WebSphere 5.1, Java 1.4 und NSE 8.2 mit den jeweils aktuellen Fixpacks wird vorausgesetzt.

Grundlage für die hier beschriebene Installation ist die MyCoRe Version 1.0 vom 15. Februar 2005.

1.Voraussetzungen für eine MyCoRe Anwendung

Zum Betrieb von MyCoRe sind zuerst einmal die folgenden freien Komponenten nötig:

1. Java SDK 1.4 oder höher (in Kombination mit CM8.2 und WebSphere wird empfohlen das WebSphere-Java zu verwenden)
2. bash (erforderlich unter Unix-Systemen für die Unixtools der Beispielanwendung)
3. cvs-Client (andernfalls muss MyCoRe unter www.mycore.de heruntergeladen werden)
4. apache-ant 1.6.x

2.Download und Installation des MyCoRe Kerns

Es wird empfohlen einen eigenen MyCoRe-Benutzer anzulegen, z.B. mcradmin. Dieser Nutzer muss Datenbankkommandos ausführen können. Dazu ist folgende Zeile in die .profile-Datei einzufügen (wobei der Benutzer db2inst1 hier exemplarisch für den db2-Standardbenutzer steht):

```
. ~db2inst1/sqlllib/db2profile
```

Nun sollte es möglich sein mit dem MyCoRe-Benutzer eine Verbindung zur DB2 herzustellen, konkret benötigt MyCoRe die Verbindung zur Datenbank des Libraryserver (standardmäßig ist dies die ICMNLSDB). Der Aufruf des Befehls:

```
> db2 connect to icmnlsdb
```

Sollte daher erfolgreich sein (weitere Hinweise siehe Punkt 3.7)

Die folgenden Schritte sind mit dem MyCoRe-Benutzer an gewünschter Stelle im Dateisystem (z.B. das Home-Verzeichnis von mcradmin /home/mcradmin) durchzuführen.

1. Download der aktuellen Version unter www.mycore.de oder via cvs


```
> cvs -d :pserver:anoncvs@server.mycore.de:/cvs checkout mycore
```

 Die .tar.gz version von www.mycore.de muß mit GnuTar ausgepackt werden, weil sonst aufgrund der langen Pfadenamen verstümmelt werden
2. Setzen der Variable MYCORE_HOME als Pfad zum MyCoRe-Verzeichnis.
(MYCORE_HOME = /home/mcradmin/mycore)
3. Für die Installation muss ausserdem ANT_HOME und JAVA_HOME (vorzugsweise auf das WebSphere-Java) gesetzt sein.
4. Anpassen der Konfigurationsdatei für die Installation,
\$MYCORE_HOME/bin/build.properties, wie folgt:
 - Auskommentieren der JDBCStore-Einträge für DB2 Version 7* und 8* (ggf. die Pfade anpassen, anstatt db2jcc.jar kann in älteren Installationen auch noch runtime.zip vorhanden sein und verwendet werden)

```
# IBM DB2 Version 7.* or 8.* under IBM AIX
MCR.JDBCStore.Type=db2
MCR.JDBCStore.BaseDir=/home/db2inst1/sqlllib
MCR.JDBCStore.Jars=java/sqlj.zip java12/db2java.zip java/db2jcc.jar
MCR.JDBCStore.ClassesDirs=function
MCR.JDBCStore.LibPath=java12 lib
```

 - Einkommentieren des JDBCStore-Eintrages für MySQL

```
# MySQL under Linux or Windows: JDBC driver is included in MyCoRe!
#MCR.JDBCStore.Type=mysql
```

- Auskommentieren der XMLStore-Einträge für Content Manager Version 8 (ggf. die Pfade anpassen)

```
# IBM Content Manager Version 8 under IBM AIX
MCR.XMLStore.Type=cm8
MCR.XMLStore.BaseDir=/opt/IBMcmb
MCR.XMLStore.Jars=lib/cmb*8*.jar lib/dtappsrv.jar
MCR.XMLStore.ClassesDirs=lib cmgmt
MCR.XMLStore.LibPath=lib
```

- Einkommentieren der XMLStore-Einträge für eXist unter Linux

```
# XML:DB eXist unter Linux
# MCR.XMLStore.Type=xmldb
# MCR.XMLStore.BaseDir=/home/mcradmin/eXist-1.0b1
# MCR.XMLStore.Jars=exist.jar lib/core/xmlrpc-1.2.jar
# MCR.XMLStore.LibPath=lib/core
```

5. Aufruf von 'ant info' in \$MYCORE_HOME, um die aktuelle Konfiguration zu überprüfen.

6. Kompilieren und packen des MyCoRe-Kerns mit dem Aufruf von 'ant jar'. Dies erzeugt in \$MYCORE_HOME/lib die Datei mycore-for-cm8.jar

Anmerkung: Weitere mögliche Befehle können mit 'ant usage' angezeigt werden.

3. Die MyCoRe Beispielanwendung

Die Beispielanwendung wird in das gleiche Verzeichnis wie der MyCoRe-Kern (im Beispiel dlwww) gespielt.

1. Download des aktuellen Sample und des DocPortals unter www.mycore.de oder via cvs
cvs -d :pserver:anoncvs@server.mycore.de:/cvs checkout docportal
cvs -d :pserver:anoncvs@server.mycore.de:/cvs checkout mycoresample

2. Setzen der Umgebungsvariablen \$DOCPORTAL_HOME und \$MYCORESAMPLE_HOME
(DOCPORTAL_HOME = /home/mcradmin/docportal
MYCORESAMPLE_HOME = /home/mcradmin/mycoresample)

3. DocPortal enthält anwendungsinstanzübergreifende Konfigurationsdateien, wie das Grundlayout der Webseiten, Datentypen etc. Fürs erste reicht die Anpassung der ContentStoreSelectionRules.xml im \$DOCPORTAL_HOME/config-Verzeichnis:

- Wenn Dokumente standardmäßig im ResourceManager gespeichert werden sollen, dann muß der Defaultwert auf CM8 gesetzt werden:

```
<!-- Default file content store, if no other rule applies: -->
<ContentStoreSelectionRules
  default="CM8"
>
```

- Einzelne Dateitypen können ebenfalls dem CM8 oder anderen Systemen zugewiesen werden (internal Filesystem FS, IBM Video Charger VC usw.)

```
<store ID="CM8">
  <type> html </type>
  <type> msppt </type>
  <type> msword95 </type>
  <type> msword97 </type>
  <type> pdf </type>
  <type> ps </type>
  <type> rtf </type>
  <type> txt </type>
  <type> xml </type>
```

```

<type> css </type>
<type> ppt </type>
<type> jpg </type>
<type> gif </type>
<type> jpeg </type>
<type> png </type>
<type> tiff </type>
</store>

```

- Den Lucene Store auskommentieren.

Anmerkung: Der Content Manager sollte die angegebenen MimeTypes unterstützen. CSS beispielsweise wird nicht standardmäßig unterstützt. Der MimeType muss angelegt werden (mittels AdminClient) wenn Dateien dieses Types vom CM verwaltet werden sollen.

4. Kompilieren anwendungsspezifischer Klassen, generieren der Schemadateien und packen des DocPortal-jars für die Beispielanwendung MyCoRe-Sample mit dem Aufruf:

```
'ant compile schema jar' in $DOCPORTAL_HOME
```

5. Konfiguration des MyCoRe-Sample, die entsprechenden Dateien bzw. Templates befinden sich in \$MYCORESAMPLE_HOME/config

- Kopieren von mycore.properties.private.template nach mycore.properties.private

- Anpassen der Datei:

1. das Passwort von icmadmin setzen

```
MCR.persistence_cm8_password=?????????
```

2. Prüfen der Pfade für die Volltextindizierung:

```

MCR.persistence_cm8_textsearch_indexdir=
    /home/db2inst1/sqllib/db2ext/indexes
MCR.persistence_cm8_textsearch_workingdir=
    /home/db2inst1/sqllib/db2ext/indexes

```

3. JDBC-Treiber für DB2 angeben, MySQL-Treiber einkommentieren:

```

# JDBC parameters for connecting to DB2
MCR.persistence_sql_database_url=jdbc:db2:ICMNLSDB
MCR.persistence_sql_driver=COM.ibm.db2.jdbc.app.DB2Driver

# JDBC parameters for connecting to MySQL
#MCR.persistence_sql_database_url=
#    jdbc:mysql://localhost/mycore?user=mcradmin&autoReconnect=true
#MCR.persistence_sql_driver=org.gjt.mm.mysql.Driver

```

4. Setzen der Host-Adresse für den Remotezugriff von anderen MyCoRe-Instanzen:

```
MCR.remoteaccess_selfremote_host=4wayb.boeblingen.de.ibm.com
```

5. Anpassen der Pfade und eMail-Adresse im Workflow-Bereich. Damit das Versenden von eMails funktioniert, muss ausserdem ein gültiger smtp-Server angegeben werden:

```

# Generic mail configuration for MCRMailer
MCR.mail.server=4wayb.boeblingen.de.ibm.com
MCR.mail.protocol=smtp
MCR.mail.debug=false

```

6. In \$MYCORESAMPLE_HOME/modules/module-wcms/aif/config die Datei

```

mycore.properties.wcms.template nach mycore.properties.wcms kopieren und
die Pfade $MycoreSampleHome und $AppsServerDocumentRoot setzen.
$AppsServerDocumentRoot =
    /usr/WebSphere/AppServer/installedApps/4wayb/mycoresample.ear/mycoresample.war/

```

7. Für das Java-Applet müssen Schlüssel generiert werden, die Konfiguration dazu geschieht mittels den "Applet JAR signing Parameters". Wichtig ist das Anpassen des Speicherortes der Schlüssel:

```
# File where the keys are stored in
SIGN.KeyStore=/home/mcradmin/mycoresample/keystore/keys
```

6. Mittels 'ant info' kann die Konfiguration nochmals angezeigt werden bzw. ein Teil davon

7. Test des db2-Zugriffs für den Benutzer, der MyCoRe installiert:

```
> db2 connect to icmnlbdb
```

Ist der Aufruf nicht erfolgreich, muss mcradmin entsprechende Rechte bekommen bzw. die Konfiguration des Benutzers angepasst werden. Dazu muss mit dem DB2-Benutzer (meist db2inst1) folgendes Kommando ausgeführt werden:

```
> db2 grant dbadm on database to user mcradmin
```

Für die nächsten Schritte muss dieser Test erfolgreich sein. Danach kann die Verbindung terminiert werden:

```
> db2 terminate
```

8. Als nächstes werden die MyCoRe-Benutzer angelegt, mit dem Aufruf 'ant create.users'
9. 'ant create.unixtools' erstellt für die weitere Arbeit mit MyCoRe hilfreichen Skripte im Verzeichnis unixtools (nur für Linux bzw. AIX, Windows-Anwender finden ähnliche Werkzeuge im Verzeichnis \$DOCPORTAL_HOME/dostools)
10. 'ant create.scripts' erzeugt das Script mycore.sh (bzw. mycore.bat) im Verzeichnis bin.
11. Die für das Ablegen der Daten im Content Manager notwendigen ItemTypes werden mit dem Aufruf von 'ant create.metastores' erzeugt. Dies kann eine Weile dauern (je nach System bis zu einer Stunde)!
12. Die Beispieldaten können nun über die MyCoRe-Kommandozeile eingestellt werden, diese ist im bin-Verzeichnis zu finden \$MYCORESAMPLE_HOME/bin/mycore.sh
Alternativ können auch die unixtools verwendet werden:

```
> cd $MYCORESAMPLE_HOME/unixtools
> ./ClassLoad.sh
> ./SAuthLoad.sh
> ./SInstLoad.sh
> ./SDocLoad.sh
```

Anmerkung: Bevor die Derivate eingestellt werden, müssen die CVS-Verzeichnisse, die beim Herunterladen von MyCoRe mittels cvs angelegt werden, gelöscht werden, da diese sonst ebenfalls eingestellt werden und es zu Fehlern kommt. Die Derivatverzeichnisse befinden sich unter \$MYCORESAMPLE_HOME/content/document/*derivate*

```
> ./SDocDerLoad.sh
```

13. Bevor die Webanwendung erstellt werden kann, muss ein Schlüsselpaar für das JavaApplet generiert werden, mittels 'ant genkeys'

14. 'ant war' erstellt eine in WebSphere installierbare Webanwendung, die auf den soeben erstellten Datenbestand zugreift

4. Installation der Webanwendung

Für die Installation der Webanwendung muss der Adminserver gestartet sein:

```
> /usr/WebSphere/AppServer/bin/startServer.sh server1
```

Läuft der Adminserver, kann im Webbrowser folgende URL für die weitere Installation aufgerufen werden: `http://<hostname>:9090/admin`

Nach der Anmeldung sind folgende Schritte durchzuführen:

1. (linke Seite) **Server Application Server**
2. (rechte Seite) **NEW**
3. (rechte Seite) Server Name **mycoresample NEXT**
4. (rechte Seite) **FINISH**
5. (linke Seite) **Applications Install New Applications**
6. (rechte Seite) **Server Path** Pfad zum File mycoresample.war eingeben **/mycoresample** im Feld Context Root eintragen **NEXT**
7. (rechte Seite) Preparing for application installation **NEXT**
8. (rechte Seite) Step 1 **NEXT**
9. (rechte Seite) Step 2 **NEXT**
10. (rechte Seite) Step 3 Auswählen **mycoresample** in der Checkbox dann auswählen der Zeile mit **server=mycoresample APPLY**
11. (rechte Seite) Step 3 **NEXT**
12. (rechte Seite) Step 4 **FINISH**
13. (linke Seite) **Server Application Server**
14. (rechte Seite) **mycoresample Process Definition Process Execution**
15. (rechte Seite) User auf **mcradmin** setzen
16. (rechte Seite) Group auf **mcr** setzen
17. (rechte Seite) **APPLY OK**
18. (rechte Seite) **Java Virtual Machine Classpath**
19. (rechte Seite) einfügen der Pfade (ggf. anpassen)


```

/home/db2inst1/sqlllib/java12/db2java.zip:
/usr/lpp/cmb/lib/cmbsdk81.jar:
/usr/lpp/cmb/cmgmt

```

Anmerkung: Ich habe hier den CLASSPATH der beim Aufruf von 'ant info' im \$MYCORESAMPLE_HOME ausgegeben wird eingetragen, da einige Klassen sonst nicht gefunden werden.

20. (rechte Seite) **APPLY OK**
21. (rechte Seite) oben auf den Text **save** klicken
22. (rechte Seite) **SAVE**
23. (linke Seite) **Environment Update Web Server Plugin OK**
24. **Logout**

Nun muss der Application Server gestartet werden:

```
> /usr/WebSphere/AppServer/bin/startServer.sh mycoresample
```

Beim ersten Mal startet der Server nicht, da einige Logdateien noch nicht geschrieben werden können. Dies liegt daran, dass die Webanwendung als **mcradmin** ausgeführt wird. Daher sind nun folgende Schritte nötig:

```
1. > chown -R mcradmin:mcr /usr/WebSphere/AppServer/logs/mycoresample
2. > mkdir /usr/WebSphere/AppServer/tranlog/mycoresample
3. > chown -R mcradmin:mcr /usr/WebSphere/AppServer/tranlog/mycoresample
4. > chown -R mcradmin:mcr
   /usr/WebSphere/AppServer/installedApps/<host>/mycoresample.ear
5. > chmod 666 /usr/WebSphere/AppServer/logs/activity.log
```

Danach ist der Server noch einmal zu stoppen und neu zu starten:

```
> /usr/WebSphere/AppServer/bin/stopServer.sh mycoresample
> /usr/WebSphere/AppServer/bin/startServer.sh mycoresample
```

Jetzt sollten Sie auf das MyCoRe-Sample unter der URL <http://<hostname>/mycoresample> zugreifen können.