

# Einführung in Managementinformationssysteme

## Übung 07

15. Dezember 2008



## Aufgabe Replizierung

- Vergeben Sie für Ihren Nachbarn Rechte zum Editieren auf Ihrer Produktdatenbank.
- Nehmen Sie einen neuen Eintrag in Ihrer Produktdatenbank vor und speichern Sie diesen ab.
- Lassen Sie diesen Eintrag danach von Ihrem Nachbarn aufrufen und editieren.
- Editieren Sie selber den Eintrag.
- Lassen Sie zuerst Ihren Nachbarn den Eintrag abspeichern. Speichern Sie danach Ihre eigene Version ab.
- Leiten Sie daraus den Repliziermechanismus ab.
- Verhält es sich bei binären Dokumenten gleich?



- Agenten sind spezielle Programme die unter Lotus Aufgaben erfüllen können
- Mit Agenten kann man Werte in Feldern verändern, Dokumente erzeugen, bearbeiten, löschen
- Sie können unabhängig von Ansichten und Masken ausgeführt werden
- Agenten können gemeinsam oder persönlich genutzt werden
- Die Agenten werden über "Shared Code – Agents – New Agent" erstellt



- Agenten können manuell oder periodisch ausgeführt werden
- Manuell können die Agenten in der Übersichtenliste mit "Run" oder "On event" von einem "View" oder einer "Form" mit @Command([RunAgent]; "CreateEntry") gestartet werden
- Der periodische Ablauf erfolgt über die "Infobox" mit "On schedule"

Agent Schedule

Run Agent More Than Once a Day

Run agent every 1 hour(s) 0 minutes

Between times 00:00 and 00:00

All day

Restrict Dates When Agent Runs

Start running agent on this date 14.01.2008 .14

Stop running agent on this date 14.01.2008 .14

Don't run agent on weekends

Where The Agent Runs

Run on Bauhaus/MIS

Choose server when agent is enabled

OK Cancel

## Rechte:

- Es sind mindestens Leser-Rechte erforderlich
- Es muß die Option "Pers. Agenten erstellen" aktiviert sein, eventuell zusätzlich die Option "Create LotusScript / Java Agents"



- Mit steigender Anzahl der gespeicherten Dokumente steigt die Ausführungszeit des Agenten
- Deswegen können Auswahloptionen in Abhängigkeit vom Typ des Agenten vorgenommen werden



Manuel aus dem Menü "Aktionen"	Alle Dokumente in der Datenbank Alle neuen und geänderten Dokumente seit dem letzten Start Alle ungelesenen Dokumente in der Ansicht Alle Dokumente in der Ansicht Ausgewählte Dokumente Einmal ausführen (@Befehle können benutzt werden)
Manuell aus der Liste der Agenten	Alle Dokumente in der Datenbank Alle neuen und geänderten Dokumente seit dem letzten Start Alle ungelesenen Dokumente in der Ansicht Alle Dokumente in der Ansicht Ausgewählte Dokumente Einmal ausführen (@Befehle können benutzt werden)
Vor Eingang neuer Mail	Jedes eingehende Mail-Dokument
Nach Eingang neuer Mail	Neu eingegangene Mails
Wenn Dokumente erstellt oder geändert wurden	Vor kurzen geänderte Dokumente
Wenn Dokumente eingefügt werden	Eingefügte Dokumente
Periodisch: Mehrmals am Tag Periodisch Täglich	Alle Dokumente in der Datenbank Alle neuen und geänderten Dokumente seit dem letzten Start



Notes bietet verschiedene Möglichkeiten für die Implementierungen an:

- Einfache Aktionen:

- Es können z.B. Werte in Feldern geändert werden, Dokumente gelöscht oder kopiert werden
- Default-Aktionen können in dieser Agentenmöglichkeit hinterlegt werden
- Beispiel: "Modify Field", "Move to Folder", "Remove from Folder"

- Formeln:

- In Formeln können die von Lotus vorgegebenen @Funktionen und @Befehle hinterlegt werden
- Beispiel: @Command([FileCloseWindow])





## ■ LotusScript

- Lotus besitzt eine eigene Programmiersprache, LotusScript
- LotusScript ähnelt stark VisualBasic
- LotusScript ist unabhängig von einer speziellen Plattform einsetzbar
- LotusScript ist eine objektorientierte Sprache
- Es werden OLE-Unterstützungen angeboten
- LotusScript ist nicht Gegenstand dieser Übung

## ■ Imported Java bzw. Java

- Die Programmiersprache Java wird ebenfalls unterstützt
- Der Code kann direkt per "Copy and Paste" eingefügt werden oder extern programmiert und dann importiert werden



- Erzeugen Sie einen Agenten der folgenden Befehl ausführt:  
`System.out.println(" Hello world" );`
- Erzeugen Sie eine Seite, die einen " Button" enthält, der diesen Agenten ausführt
- Lassen Sie sich beim Ausführen den Text " Hello world" auf der Java-Konsole (" File - Tools - Show Java Debug Console" ) anzeigen



- Erzeugen Sie eine neue Datenbank mit dem Namen Ihrer Immatrikulationsnummer unter "Students\_0708/Abschluss" auf dem Lotus-Server
- Erzeugen Sie eine "Form" in Tabellenform mit den Feldern "Vorname", "Strasse", "Telefonnummer"
- Erzeugen Sie einen "View", der Ihnen diese Felder anzeigt
- Machen Sie einen Probearbeit
- Speichern Sie die Datei `http://bauhaus.cs.uni-magdeburg.de/wikitm/0809/telefonbuch.txt` auf Ihrem Computer ab
- Erzeugen Sie einen Java-Agenten, der Ihnen die semikolonseparierten Werte in die erzeugte Notes Datenbank importiert
- Legen Sie in der "Form" einen Ausführknopf an, der Ihnen diesen Agenten ausführt



## Hinweis Java-Code: Auslesen der Datei:

```
import java.io.BufferedReader;
import java.io.FileReader;
import java.util.StringTokenizer;
.....
String line;
BufferedReader f = new BufferedReader(new FileReader("C:Pfad_zur_txt_datei"));
StringTokenizer st = null;
while ((line = f.readLine()) != null) {
st = new StringTokenizer(line,";");
while(st.hasMoreTokens()) { //denken Sie daran: st.nextToken() ist der Vorname,
etc.
} }
f.close();
```



## Hinweis Java-Code: Einfügen von Datensätzen:

```
Database db = agentContext.getCurrentDatabase();  
View view = db.getView("Name_Ihres_Views");  
Document destinationDocCopy = db.createDocument();  
destinationDocCopy.save();  
lotus.domino.Document destinationDoc =  
view.getLastDocument();  
destinationDoc.replaceItemValue("Vorname", "Mein erster  
Vorname");  
destinationDoc.save();
```

