Multimedia im Netz –
Wintersemester 2011/12

Übung 10

Medien Informatik München

Betreuer: Sebastian Löhmann
Verantwortlicher Professor: Prof. Dr. Heinrich Hussmann

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Organisatorisches

Gesundes neues Jahr ☺

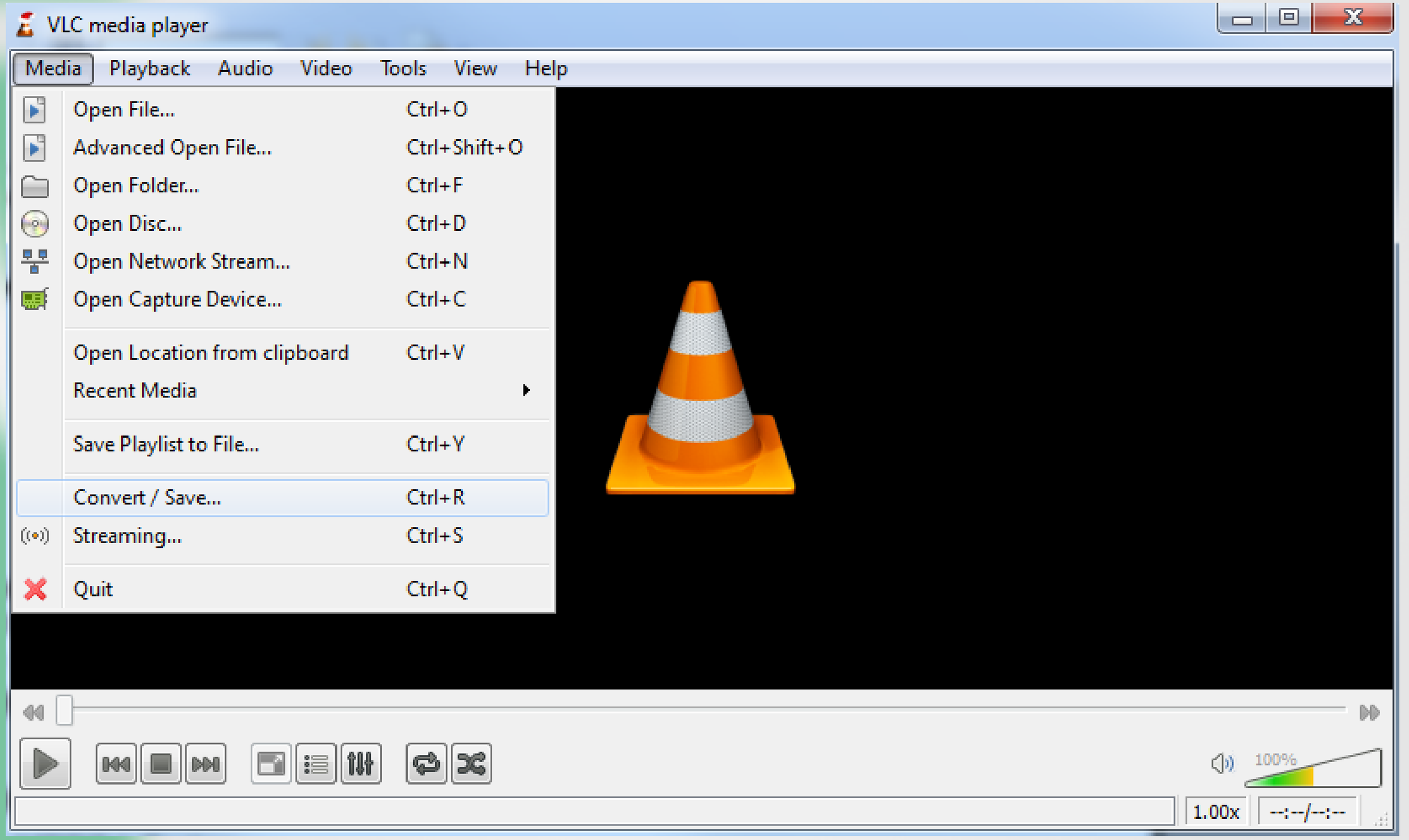
Blatt 08

Videoformate im Netz

Blatt 08: Videoformate im Netz

Browser	Formats supported by different web browsers		
	Ogg Theora	H.264	VP8 (WebM)
Internet Explorer	Manual install[note 2]	9.0[23]	Manual install[note 3]
Mozilla Firefox[26]	3.5[28]	Manual install (Only on Windows)[note 4]	4.0[30][31]
Google Chrome	3.0[32][33]	removed after 10.0[34][*not in citation given*]	6.0[35][36]
Chromium	r18297[37]	Manual install[note 5]	r47759[39]
Android browser	2.3[40]	3.0[41]	2.3[42]
Safari with Quicktime	Manual install[note 6]	3.1	Manual install[43]
Opera	10.50[44]	No	10.60[45][46]
Konqueror	4.4[48]	Manual install[note 7]	Yes[50]
Epiphany	2.28[52]	Manual install[note 8]	Yes[note 8][53]

Blatt 08: Videoformate im Netz



JSF

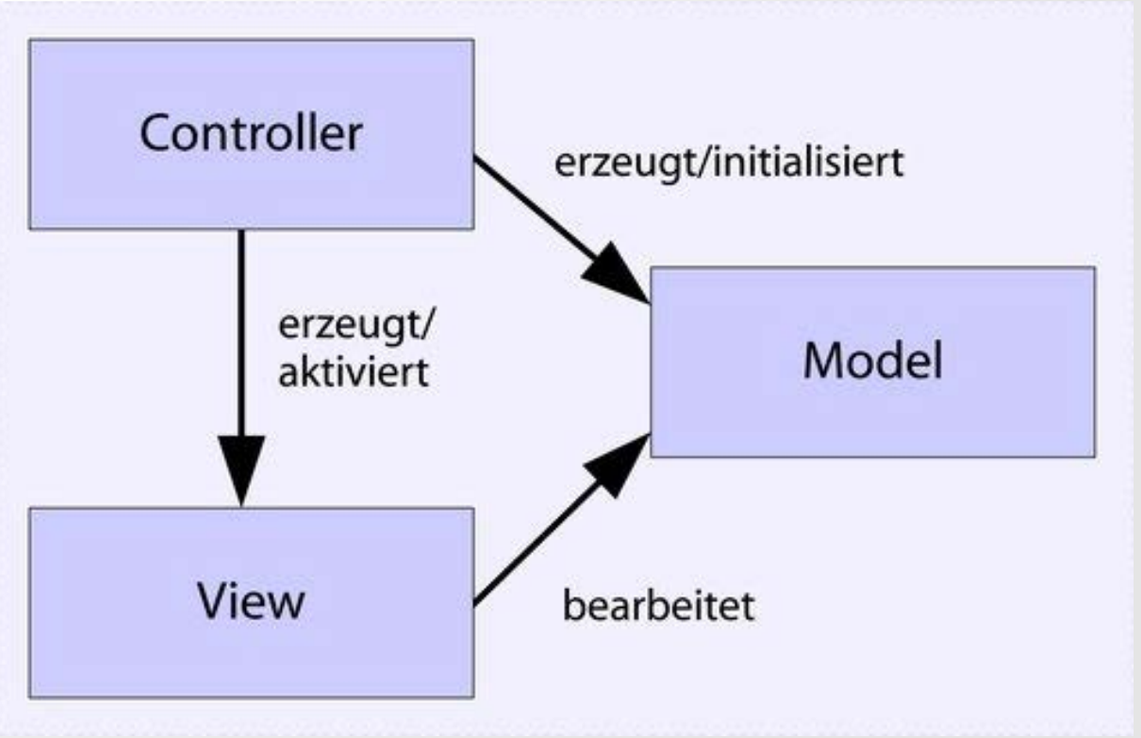
Java Server Faces

JSF: Intro

- Java Framework für Web-Anwendungen
- aktuelle Version ist 2.0
- Ziel: Trennung von Java-Code und Markup
- folgt der Model-View-Controller-Architektur
- nächste Woche mehr dazu

Model-View-Controller-Architektur

- View
  - Präsentation, Formulare, … (z.B. HTML)
  - nimmt Nutzereingaben entgegen (Observer-Pattern)
- Model
  - Datenmodell, Programmlogik (z.B. Java)
  - Konstruktor, Getters, Setters
- Controller
  - Programmsteuerung
  - bearbeitet Nutzereingaben



Ziel in dieser Woche

- Einrichten einer Umgebung, in der wir JSF-Anwendungen erstellen und testen können
- Dazu benötigen wir
  - Programmiersprache JAVA (JavaBeans)
  - Entwicklungsumgebung (IDE) Eclipse
  - Applikationsserver JBoss
  - JBoss-Tools-Plugin für Eclipse
  - Zugangsdaten für den Server
  - SFTP client für Kommunikation mit dem Server
  - Browser ☺

Java

- hat natürlich schon jeder installiert ☺
- Falls nicht: Version Java Platform (JDK) 7u2
- http://www.oracle.com/technetwork/java/javase/downloads/index.html
- Nicht vergessen: Environment Variable JAVA_HOME setzen (googeln)

Eclipse

- Entwicklungsumgebung (IDE)
- http://www.eclipse.org/downloads/packages/release/indigo/r
- Eclipse IDE for Java EE Developers
- Hinweis: Eclipse als Admin starten (Windows), um mit remote Server kommunizieren zu können

JBoss (Standalone)

- Applikationsserver für den eigenen Rechner
- wichtig zum testen/probieren zu Hause
- dort wird später unsere Anwendung „deployed“ und über einen Browser aufgerufen
- JBoss Version AS 7.0.2
- http://www.jboss.org/jbossas/downloads/
- Entpacken
- bin\standalone.(bat/sh) starten (als Admin)

JBoss-Tools für Eclipse

- Einbinden und Kommunikation mit JBoss-Server in Eclipse
- Infos unter http://www.jboss.org/tools/
- Installation über Eclipse Marketplace (zu finden unter „Help“ im Eclipse-Menü)
- JBoss Tools Indigo verwenden

Erstes JSF-Projekt anlegen

- In Eclipse “Dynamic Web Project” erstellen
- Target Runtime “JBoss 7.x” auswählen
- Neuen lokalen Server erstellen, dafür lokales JBoss-Verzeichnis auswählen
- bei der Konfiguration (modify) JSF auswählen!
- web.xml-Datei automatisch erstellen lassen

Erste JSF-Testanwendung erstellen

- Eclipse: “Web Development”-Perspektive wählen
- Im unteren Teil ist nun lokaler Server zu finden (kann zB gestartet und gestoppt werden)
- zip von der Website laden
- Java-File in den src-Folder kopieren
- XHTML-Files in den WebContent-Folder kopieren
- web.xml-File in den WebInf-Ordner kopieren
- Projekt als war-File nach server\standalone\deployments exportieren
- Projekt per drag&drop auf Server ziehen

Anwendung starten

- war-File wird vom Server automatisch erkannt, Anwendung wird “deployed”
- Browser öffnen
- http://localhost:8080/HelloJSF/hello.jsf
  (Anpassung der Datei-Endung wird vom Server aufgrund der Infos in web.xml übernommen!)
- Anwendung sollte nun funktionieren

JBoss (remote)

- Applikationsserver, den wir remote zur Verfügung stellen
- Hier sollen später die erstellten Anwendungen laufen
- fertige Anwendungen können per SFTP auf den Server übertragen werden
- Beispiel für Windows-SFTP-Client: WinSCP

Zugangsdaten

- Zugangsdaten zum Server können per Mail bei Sebastian Löhmann erfragt werden

Warnung

- Alle haben denselben Account auf dem Server
- Bitte Projektnamen verwenden, der noch nicht vergeben ist
- Bitte keine anderen Deployments löschen ☺ (wir führen log files mit Zugriffen…)
- Source-Code nicht in das war-File integrieren, da der Code sonst eingesehen werden kann

SFTP-Plugin für Eclipse

- als Alternative zur SFTP-Anwendung kann man auch SFTP in Eclipse nutzen
- Plugin „Target Management“
- http://www.eclipse.org/tm/
- Nach Installation steht neue View in Eclipse zur Verfügung: „Remote Systems“
- jetzt kann neuer Remote-Server (ssh only) erstellt werden, sftp-Funktionalität wird automatisch zur Verfügung gestellt

Blatt 09

Tutorial erstellen

- die beschriebenen Schritte durchführen bis Anwendung auf dem Server ausgeführt werden kann
- den Weg dorthin dokumentieren (Stichpunkte und Screenshots)
- Tutorial erstellen nach dem Motto:

  “JSF HelloWorld mit JBoss und Eclipse”
- wahlweise auf deutsch oder englisch

Ziele der Übung

- Funktionierende Umgebung erstellen, sodass nächste Woche programmiert werden kann (Voraussetzung für Blatt 10)
- Tutorial erstellen, dass man später selbst verwenden kann
- Tutorial, dass von anderen verwendet werden kann (z.B. online oder für die Übungen nächstes Jahr)
- Viel Spaß und viel Erfolg!

Danke ☺ Fragen?

Medien Informatik München

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN