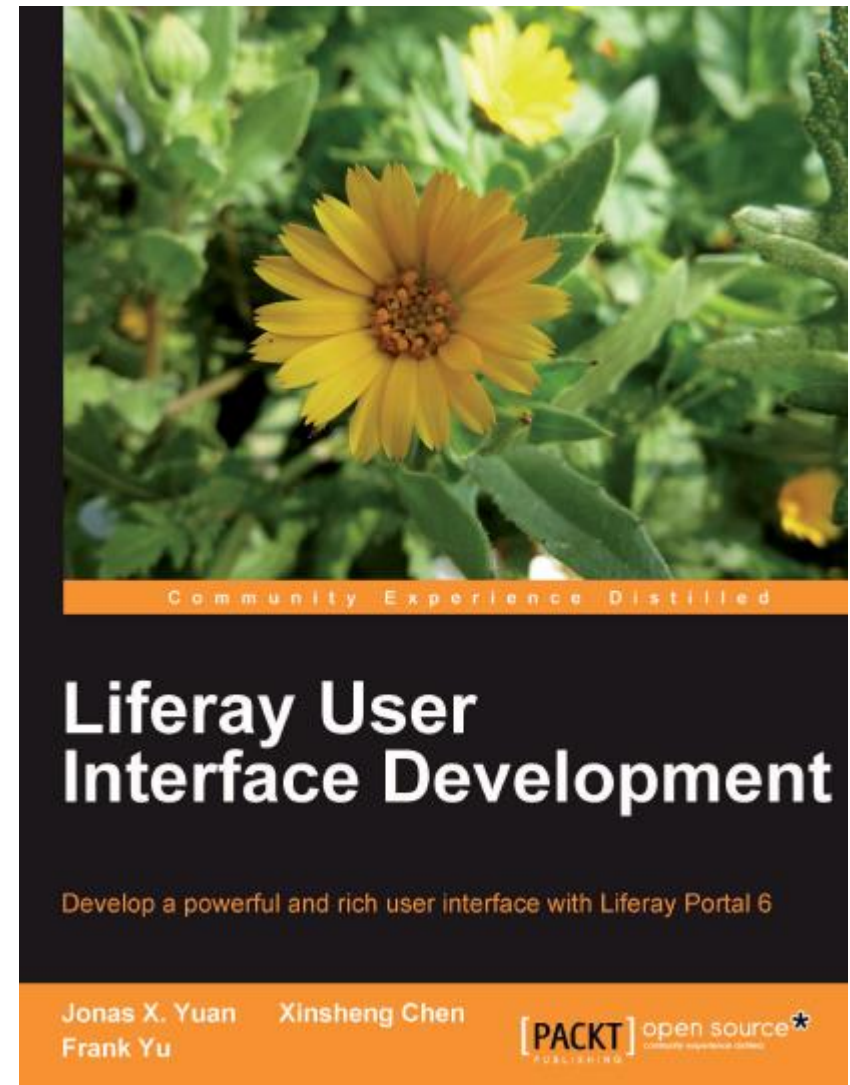
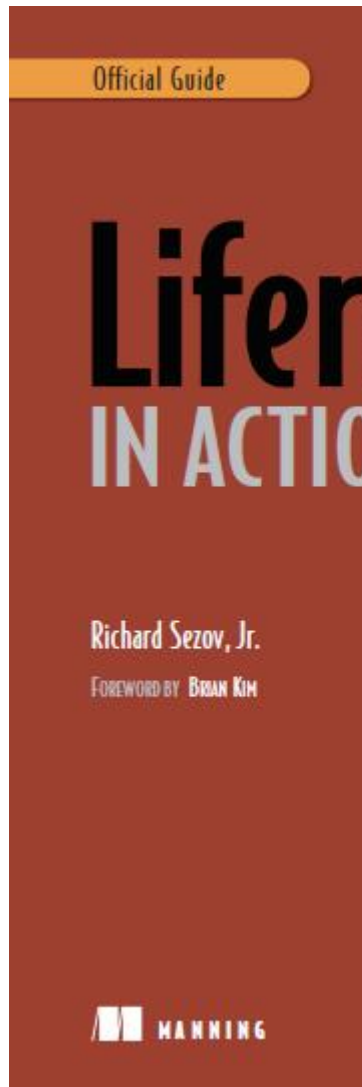


Liferay

Ein Übersichts-Workshop für BITBW



- Dies ist ein praktisches Seminar
 - Übungen vertiefen die vermittelten Inhalte
 - Ergänzend Diskussion, Demonstrationen
- Die in diesem Seminar verwendete Werkzeuge und Frameworks sind Open Source
 - LPGL Lizenzmodell
- Dokumentation und Ressourcen stehen auch im Internet zur Verfügung
 - Insbesondere die API-Dokumentation
- Konventionen
 - Befehle werden in `Courier-Schriftart` dargestellt
 - Dateinamen werden in *kursiver Courier-Schriftart* dargestellt
 - Links werden in unterstrichener Courier-Schriftart dargestellt

© Javacream

Javacream

Dr. Rainer Sawitzki

Alois-Gilg-Weg 6

81373 München

eMail: training@rainer-sawitzki.de

Alle Rechte, einschließlich derjenigen des auszugsweisen Abdrucks, der fotomechanischen und elektronischen Wiedergabe vorbehalten.

Einführung	6
Liferay Anwender	22
Liferay-Programmierung	50

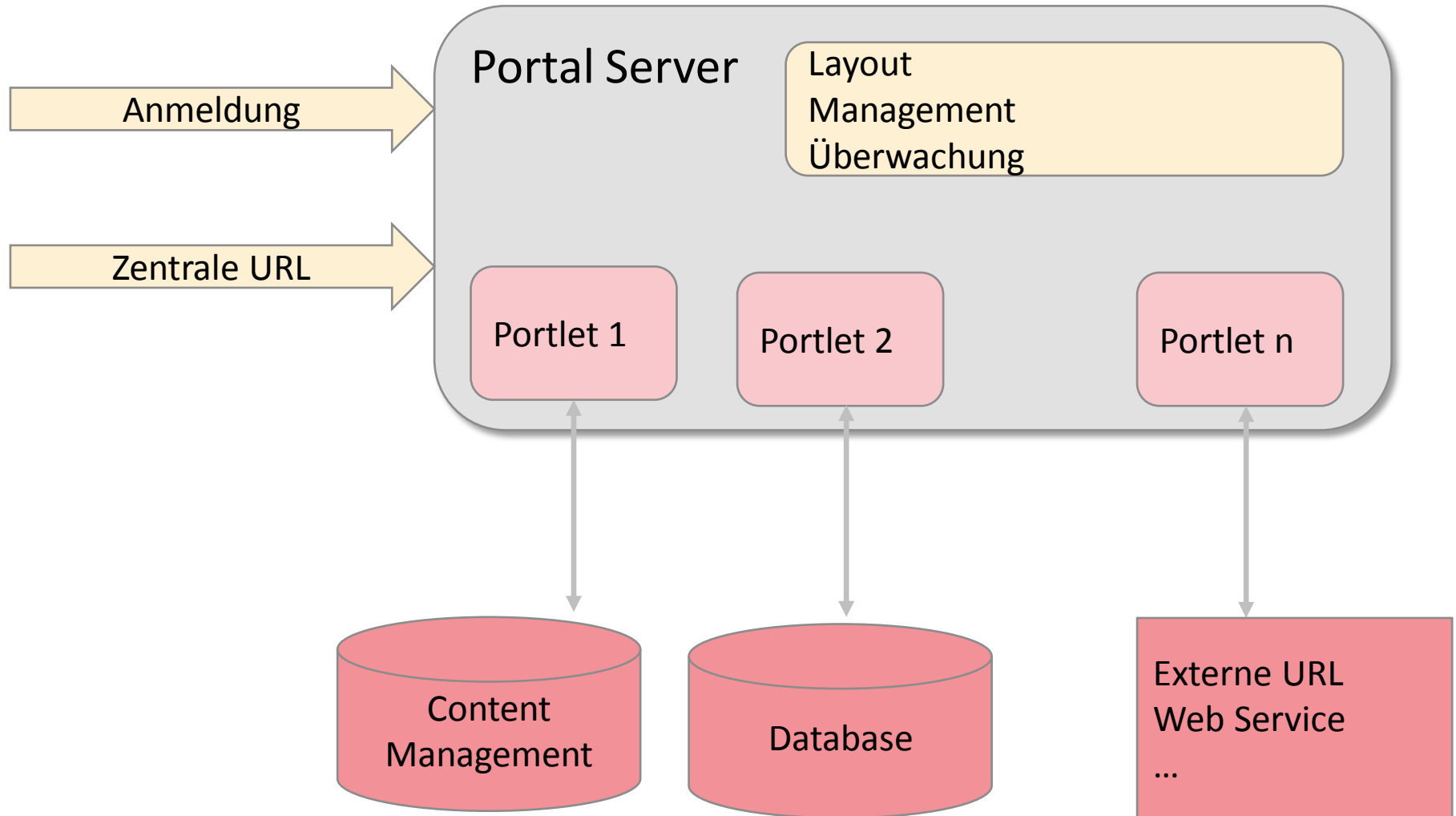
1

EINFÜHRUNG

1.1

ARCHITEKTUR

- "Ein Portal-Server ist eine Integrations-Plattform für grafische User-Interfaces"
 - Komposition aus unabhängigen Komponenten
 - Portlets
 - Portlets liefern HTML-Fragmente
 - Diese werden vom Portal-Server in die Gesamtseite integriert
 - Bus-System zur Inter-Portlet-Kommunikation
- Zentrale Dienste
 - Authentifizierung
 - Autorisierung



- Eine Java-basierte Implementierung eines Portal-Servers
 - Läuft innerhalb einer Java Virtual Machine
- Präziser: Eine JEE-Anwendung
 - Liferay kann in jedem üblichen Applikationsserver installiert werden
 - Deployment als
 - .war
 - .ear
- Zur Server-Überwachung können übliche Verfahren eingesetzt werden
 - Beispielsweise JMX
 - HAWTIO
 - Nagios
 - ...

- Liferay Portal
 - JSR-168/JSR 286
- Liferay CMS and WCM
 - JSR-170 Content Management System mit Web-Zugriff
- Liferay Collaboration
 - Blogs, Wiki, Kalender
- Liferay Social Office
 - Arbeitsgruppen und Team-Zusammenarbeit

- Die Liferay-Distribution enthält eine Vielzahl fertig einsetzbarer Portlets
 - Content-Management
 - Content-Visualisierung
 - Proxies für externe URLs
 - Social Media
- Ausgefeiltes Berechtigungskonzept
- Etablierte Java-Frameworks sind integriert
 - JavaServer Faces
 - Vaadin
 - Spring
 - JPA/Hibernate

1.2

INSTALLATION UND WERKZEUGE

- Distribution als
 - Reine Web-Anwendung
 - Gebündelt mit Applikationsserver
- Download des Liferay SDKs
 - Projekt-Generatoren
 - Build-Skripte
- Optional: Installation zusätzlicher Komponenten
 - Liferay Studio
 - Ein Eclipse-PlugIn zur Entwicklung von Liferay-Anwendungen
 - Liferay Faces
 - JavaServer Faces Provider
 - Alloy UI
 - Auf YUI basierende JavaScript-Bibliothek
 - Liferay Mobile



Liferay Social Office

Liferay Sync

Liferay Projects

Dokumentation

Community

Marketplace



Liferay Roadshows
See Liferay Portal in person!

See 6.2 in Action!

Learn why Gartner positions Liferay
as a Leader for Horizontal Portals

[Read the New Report >](#)

Produkte

Entscheider ▾

Entwickler ▾

Über uns



Community



Bundled with Tomcat

Bundled with Geronimo

Bundled with Glassfish

Bundled with JBoss

Bundled with Jetty

Bundled with JOnAS + Tomcat

Bundled with Resin

Portal JavaDocs

Portal Source

Portal SQL Scripts

Plugins SDK

Bundled with Tomcat

Enterprise

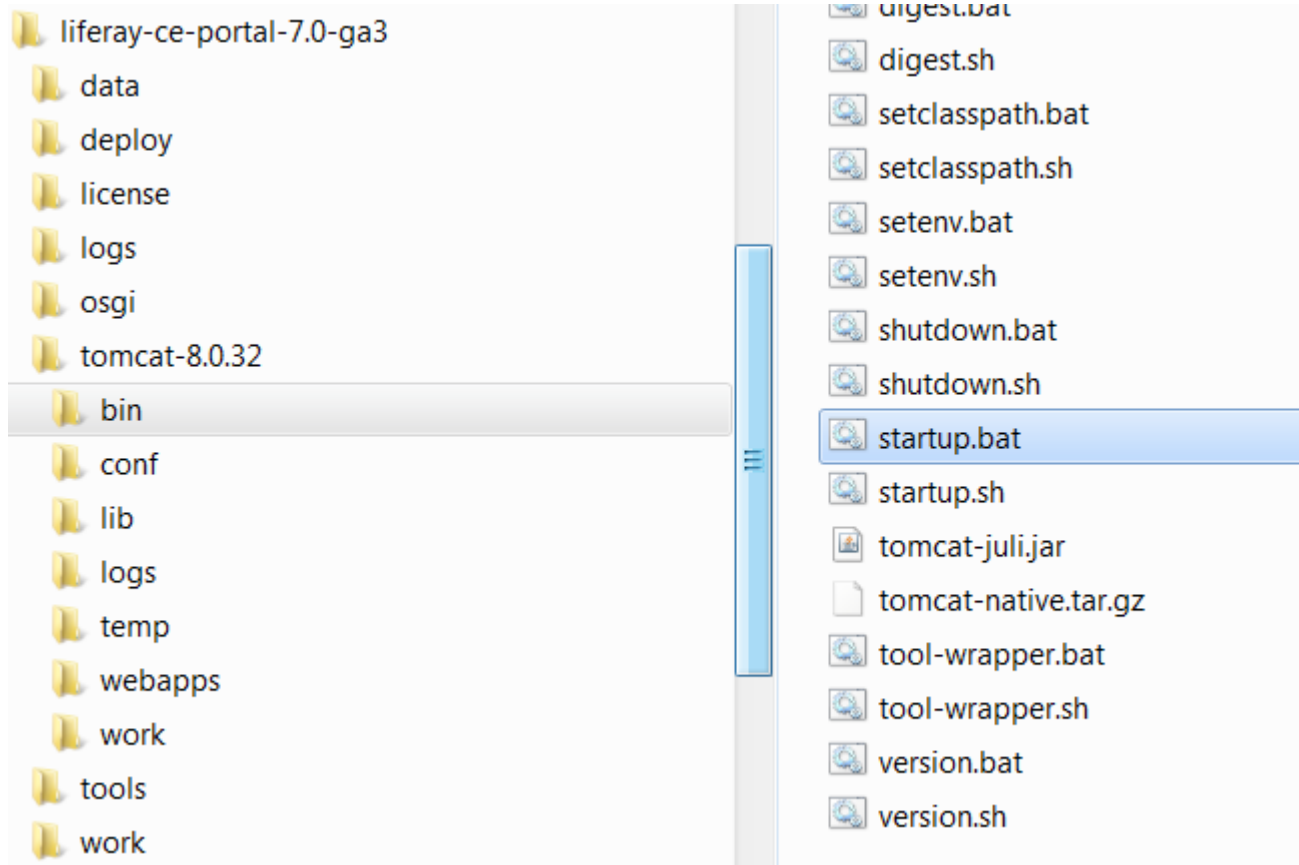


- Stabile Releases mit Bug-Fixes
- Geeignet für geschäftskritische Installationen
- Verwendbar zur Entwicklung von Produktivprojekten
- Teil der [Liferay Enterprise Subscription](#)

Liste unserer [Third Party Software hier](#)

1.3

ERSTES ARBEITEN MIT LIFERAY





⚙ Basic Configuration

Portal

Portal Name

For example, Liferay.

Default Language

 ▼

Change

☒ Add Sample Data

Administrator User

First Name

Last Name

Email (Required)

Database

Default Database (Hypersonic)

This database is useful for development and demo'ing purposes, but it is not recommended for production use.

[\(Change\)](#)

Finish Configuration



Liferay

Sign In

Welcome

Search



Main Page

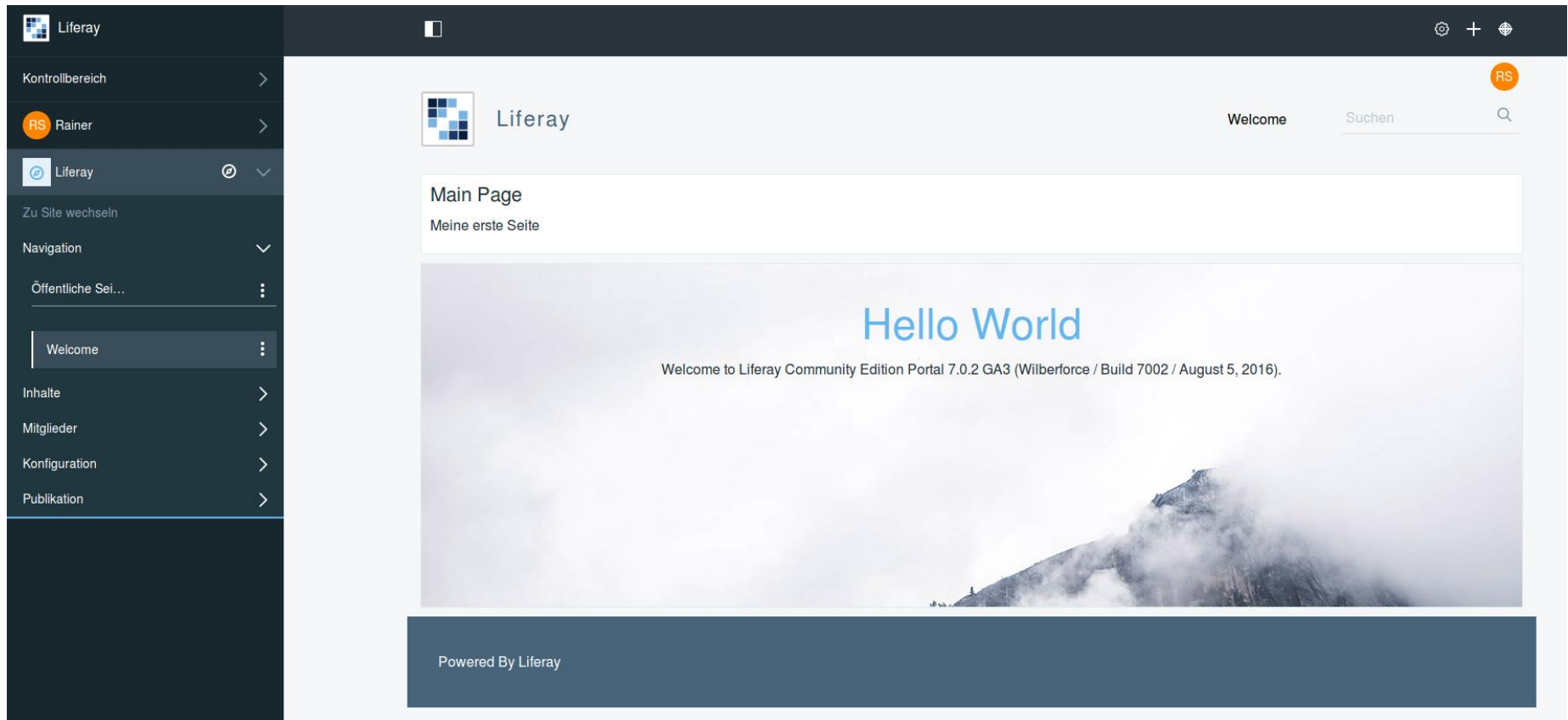
Meine erste Seite

Hello World

Welcome to Liferay Community Edition Portal 7.0.2 GA3 (Wilberforce / Build 7002 / August 5, 2016).

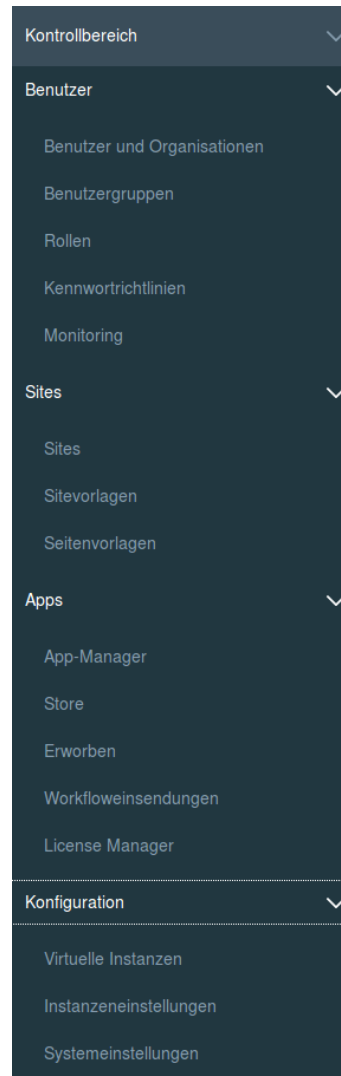
Powered By Liferay

Startseite mit Admin-Anmeldung



The screenshot displays the Liferay Admin Portal interface. On the left is a dark sidebar with navigation links: Liferay, Kontrollbereich, Rainer, Liferay, Zu Site wechseln, Navigation, Öffentliche Sei..., Welcome, Inhalte, Mitglieder, Konfiguration, and Publikation. The main content area features the Liferay logo, a 'Welcome' message, a search bar labeled 'Suchen', and a 'Main Page' section titled 'Meine erste Seite'. The central banner reads 'Hello World' and 'Welcome to Liferay Community Edition Portal 7.0.2 GA3 (Wilberforce / Build 7002 / August 5, 2016)'. The footer states 'Powered By Liferay'.

■ Kontrollbereich



2

LIFERAY ANWENDER

2.1

GRUNDPRINZIPIEN DER BEDIENUNG

- Liferay 7 ist eine typische Rich Internet Application
 - Dynamisches Ein- und Ausblenden von Bedienelementen
 - Dialoge
 - Menüstrukturen
- Benutzerführung auch auf Mobilgeräte ausgerichtet

- Ein- und Ausklappbares Menü



- Seite konfigurieren



- App oder Content hinzufügen



- Simulation verschiedener Endgeräte



- Menü



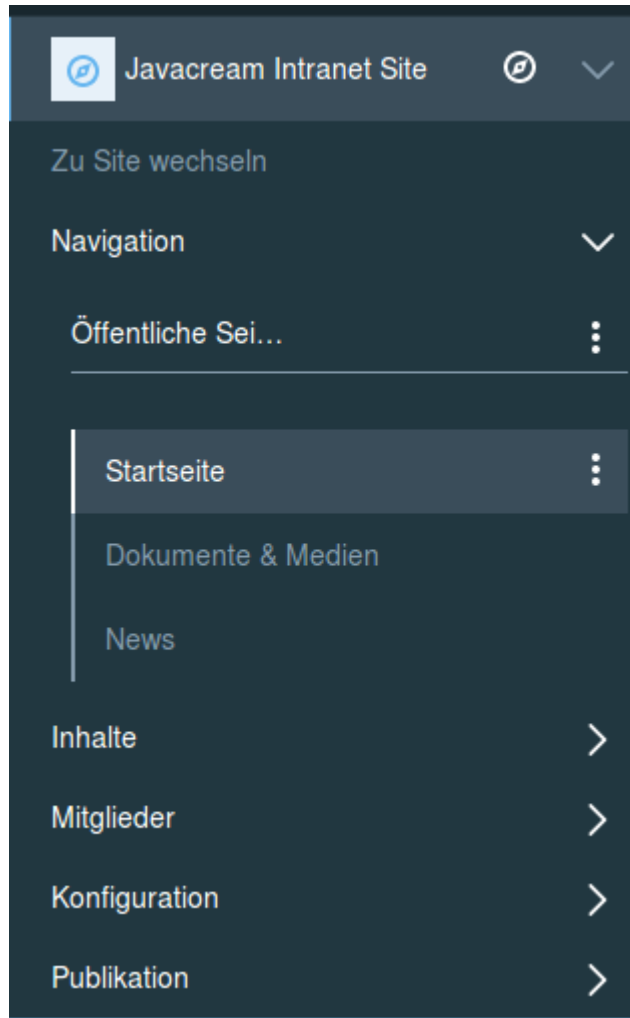
- Aufklappbare Liste




2.2


SITES, PAGES, APPS

- Rollen
 - Owner
 - Administrator
 - Member
 - Guest
- Autorisierung
 - Owner können die Site beliebig ändern und Benutzer-Rechte vergeben
 - Administratoren können Pages erstellen und verändern
 - Member können die Site benutzen
 - Also beispielsweise Content bearbeiten
 - Guests haben meistens nur Leseberechtigung
- Ein Container für beliebige viele Pages




- Sind Container für Apps
 - Der Site-Administrator kann Apps in einem auswählbaren Seitenlayout beliebig auf der Seite platzieren
- Können Unterseiten enthalten
- Werden in einem Navigationsmenü angezeigt

 **Javacream Intranet Site**

[Startseite](#) [Dokumente & Medien](#) [News](#) [Suchen](#) 

Aktivitäten

[RSS](#) 


Januar 15

TN

08:39


Teil Nehmer1 hat das Dokument 1-ApacheCamelStadtMünchen - Tabelle1.pdf hochgeladen. Datei herunterladen Zu Ordner wechseln

Sprachauswahl



Asset Publisher

[1-ApacheCamelStadtMünchen - Tabelle1.pdf](#)

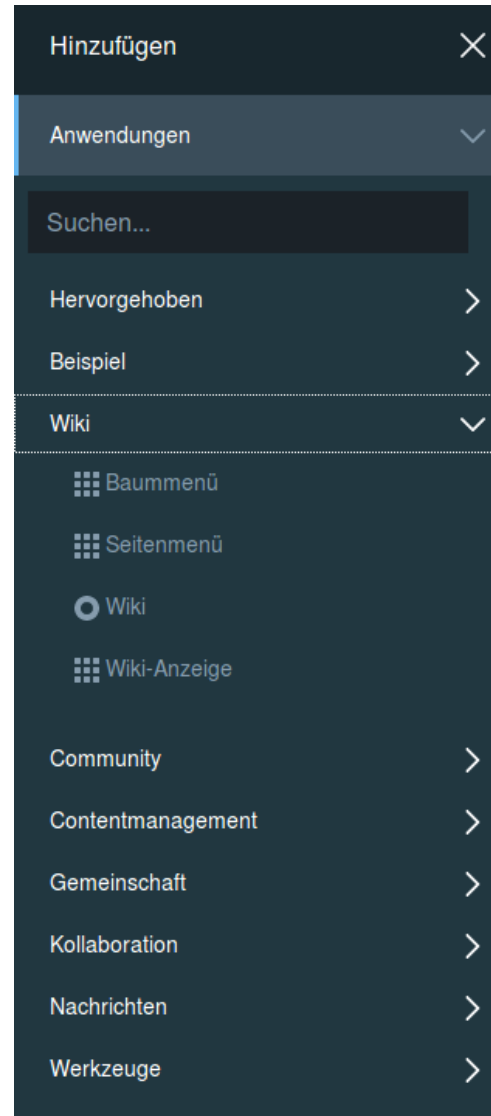


[Test](#)

- Dekorationen ermöglichen die Konfiguration und Platzierung einer App



■ Hinzufügen einer App

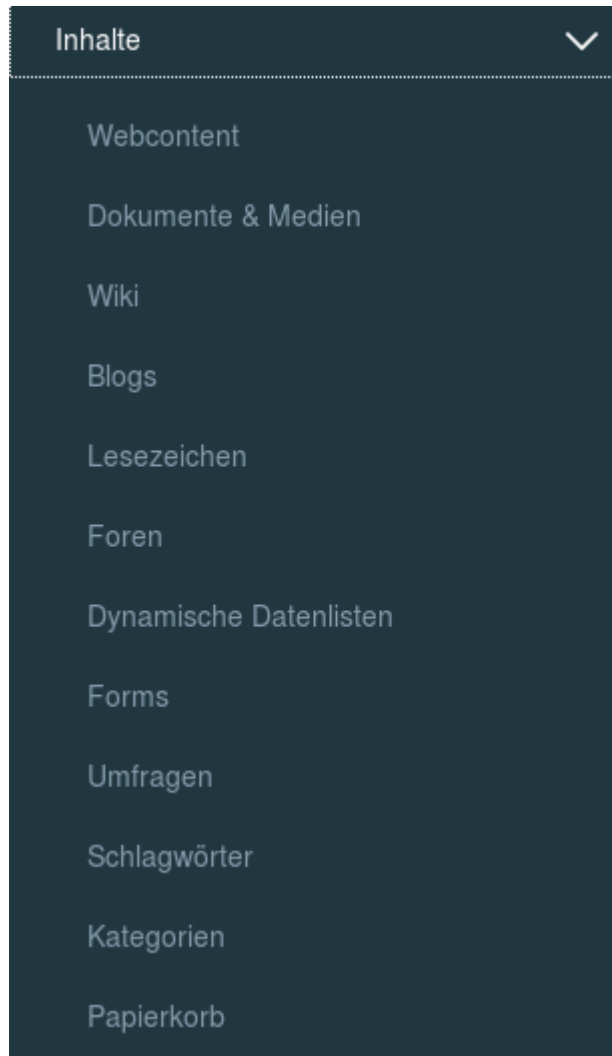


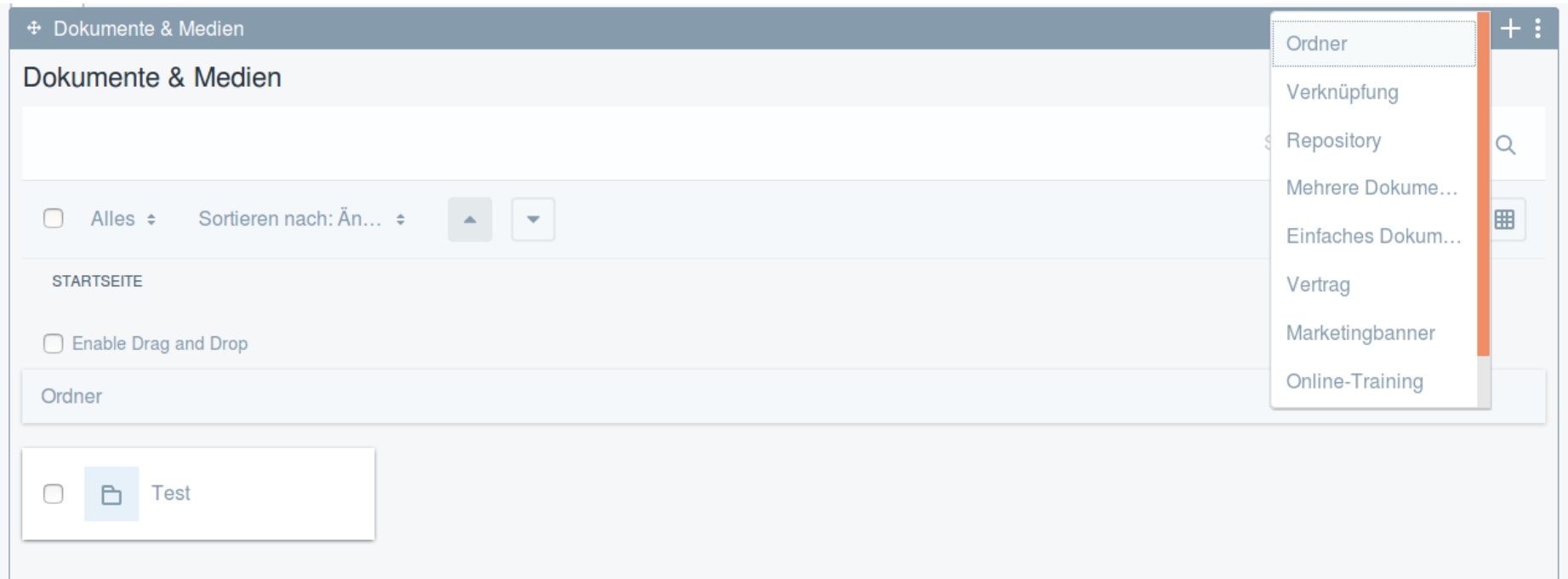
- Apps stellen prinzipiell beliebige Funktionalität zur Verfügung
- Die Standard-Distribution enthält eine Satz fertiger Apps
- Weitere Apps
 - Marketplace von Liferay
 - Eigene App-Entwicklung
 - Portlets
 - Ab Liferay 7 als OSGi-Bundle installierbar

2.3

CONTENT

- Liferay ist **auch** ein CMS
- Folgende Funktionen werden unterstützt
 - Organisation von Dokumenten
 - Tagging
 - Versionierung
 - Suche
- Zusätzlich aber auch
 - Erfassung
 - Integrierter Editor
 - Formulare
 - Collaboration
 - Kommentare
 - Bewertungen
 - Publishing und Staging





The screenshot shows the 'Dokumente & Medien' (Documents & Media) interface. At the top, there's a header bar with a plus icon and the text 'Dokumente & Medien'. Below this, the main area is titled 'Dokumente & Medien'. There's a search bar with a magnifying glass icon. Below the search bar, there's a section labeled 'STARTSEITE' (Start Page) with a checkbox for 'Enable Drag and Drop'. Below that, there's a section labeled 'Ordner' (Folders) with a checkbox and a folder icon next to the text 'Test'. A context menu is open on the right side of the interface, showing a list of options: 'Ordner', 'Verknüpfung', 'Repository', 'Mehrere Dokume...', 'Einfaches Dokum...', 'Vertrag', 'Marketingbanner', and 'Online-Training'.

✚ Dokumente & Medien


Dokumente & Medien

☐ Alles ▾ Sortieren nach: Än... ▾ ▲ ▼

STARTSEITE

☐ Enable Drag and Drop

Ordner

☐  Test

- Ordner
- Verknüpfung
- Repository
- Mehrere Dokume...
- Einfaches Dokum...
- Vertrag
- Marketingbanner
- Online-Training


- Arten von Content
 - Webcontent
 - HTML-basierter Content
 - Dokumente&Medien
 - Klassisches CMS mit beliebigen Datentypen
 - Wiki
 - Blog
- Allen Contents können Meta-Daten zugeordnet werden
 - Kategorisierung
 - Assets
 - Bilder etc.

Beispiel: Webcontent mit Rich Editor

Standardsprache: 🇺🇸 Englisch (Vereinigte Staaten von Amerika) + Übersetzung hinzufügen ▾

Titel ✱
Demo Content

Zusammenfassung
Einfacher Demo Content




Content

Dieser einfache Demo Content soll

Suchbar
☒ JA

Struktur und Vorlage ▾

Struktur: Einfacher Webcontent



Laden Sie Dokumente hoch, die eine Größe von 100MBk nicht überschreiten.

Datei

Durchsuchen...

Keine Datei ausgewählt.

Titel *

Vorhandener Content

Beschreibung

Kategorisierung

Schlagwörter

Hinzufügen

Auswählen

Verknüpfte Assets

- Hierzu dient z.B.
 - der Asset-Publisher
 - Letzte Aktivitäten
 - Bewertungen...



Asset Publisher - Konfiguration

Setup Kommunikation Teilen

Asetauswahl Darstellungsoptionen Abonnements

☒ Dynamisch
☐ manuell

Quelle >
Filter v

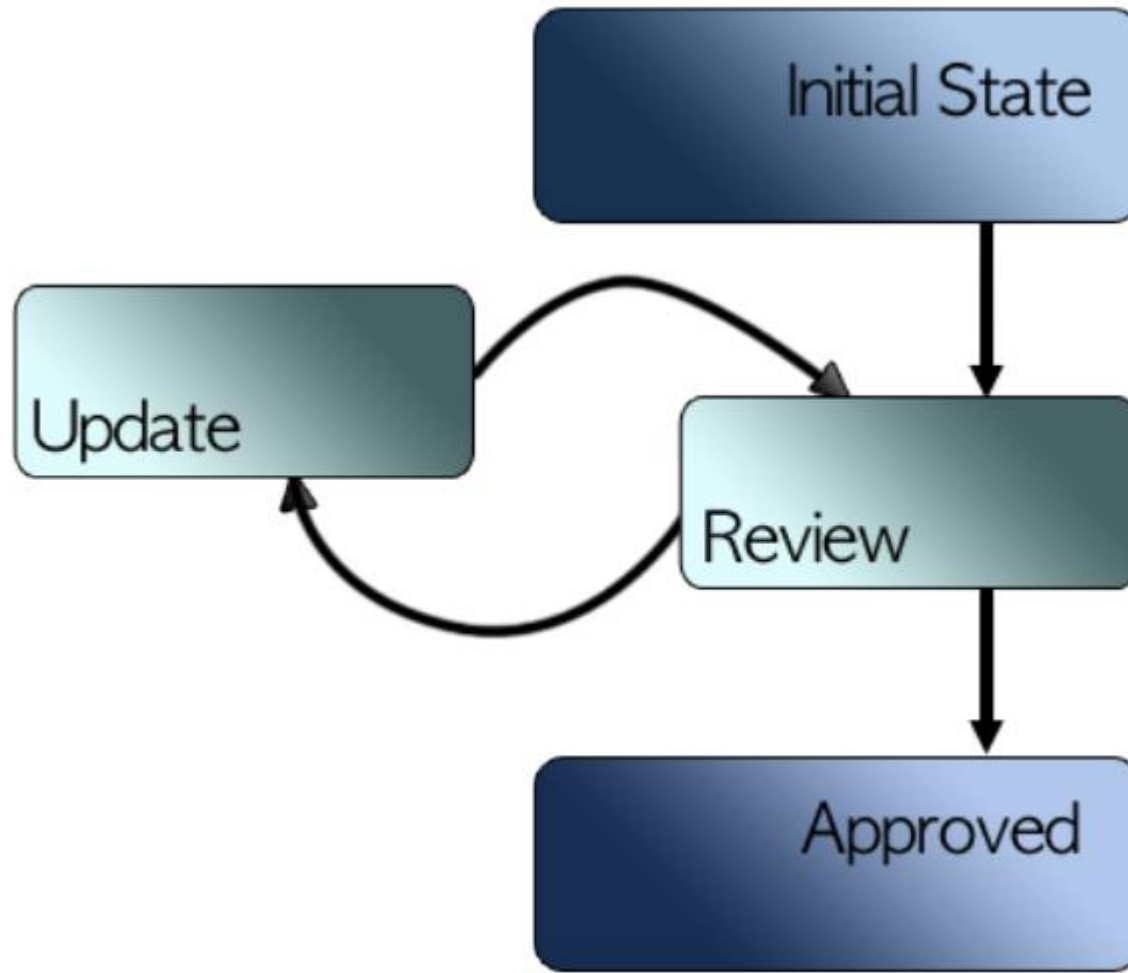
Angezeigte Assets müssen dieser Richtlinien entsprechen.
Enthält ▾ Beliebige(r) ▾ vom folgenden Schlagwörter ▾

Hinzufügen Auswählen

+ -

Nur Assets mit *Scratch* als Anzeigeseite anzeigen.
☐ NEIN
Schlagwörter aus der URL berücksichtigen.
☒ JA
Benutzerdefinierte Attribute >

- Liferay implementiert eine Workflow-Engine
 - Ein Workflow besteht aus Zuständen und Übergängen
- Simpler Flow
 - Anlegen eines Dokuments
 - Draft
 - Publish
- Workflows-Übergänge können auch zeitgesteuert (ohne Benutzer-Interaktion) ausgeführt werden
 - Automatisches Publishing eines Dokuments zu frei definierbaren Zeiten
- Zusätzliche Workflows
 - Liferay Marketplace
 - Eigene Workflows können entwickelt werden



2.4

ANPASSUNGEN

- Jede Page hat ein Layout
 - Die Apps der Page werden gemäß des Layouts angeordnet
- Eine Site kann ein Theme definieren
 - Im Wesentlichen das allgemeine Design der Site
 - Logos, Header, Footer
 - Farben und Fonts

- Jede Seite kann einen eigenen Satz von Design-Regeln enthalten
 - CSS = Cascading Style Sheets
- Der Designer muss in der Liferay-Dokumentation die geeigneten CSS-Selektoren finden
 - "Was muss ich tun, um die Hintergrund-Farbe der Seite zu ändern"
 - Viele Apps bringen ihren eigenen Satz von CSS-Selektoren mit
 - "Wie kann ich die Darstellung des Asset-Publishers ändern?"

- Bei der Erfassung von Content werden Meta-Daten erfasst
- Liferay definiert einen Standard-Satz von Eigenschaften
 - Titel
 - Abstract
 - Keywords/Assets
 - ...
- Benutzerdefinierte Meta-Daten sind möglich
 - und im Unternehmen durchaus üblich

- Die Datenerfassung von Webcontent erfolgt in einem Freitext-Editor
 - Im Endeffekt wird hier blankes HTML benutzt
- Strukturierter Webcontent kann einfacher erfasst werden:
 - Benutzung eines Templates
 - Freitext mit Platzhaltern
 - Freemarker und Velocity sind hier bekannte Sprachen
 - Eingabe der Daten über ein Formular
 - Der Benutzer gibt damit nur noch Daten ein
 - Das HTML des Content wird über das Template gerendered.

3

LIFERAY-PROGRAMMIERUNG

3.1

ÜBERSICHT

- Java-Entwickler können Liferay-Apps selber schreiben
 - Und damit die Funktionen von Liferay komplett erweitern
 - Liferay ist damit ein Applikationsserver
- Technologie-Stack:
 - Java
 - Spring
 - JavaServer Faces
 - Java Persistence API
 - Liferay-API
 - "UserLocalServiceUtil"
- Die Installation einer App erfolgt über ein Web-Archiv

- Liferay 7 ist eine komplette Neuentwicklung des Servers auf Basis von OSGi
 - OSGi ist eine Java-Plattform, die dynamisch Services zur Verfügung stellt
 - "OSGi-Bundles"
 - Standard-Services wie Logging, Events/Notifications
 - Das OSGi-API ermöglicht ein dynamische Registrieren bzw. einen Lookup von Services
 - Definition von Bundle-Abhängigkeiten durch ausgefeilten Import/Export-Mechanismus
- Damit können Liferay-Erweiterungen auch als OSGi-Bundles installiert werden

- Liferay-Administratoren können über eine Script-Konsole direkt Programm-Sequenzen ausführen lassen
 - Standard-Sprache: Groovy
 - Eine Art Skriptsprache für Java

```
ServiceContext serviceContext = new ServiceContext();
    user = UserLocalServiceUtil.addUser(
        creatorUserId, companyId, autoPassword, password1,
password2,
        autoScreenName, screenName, emailAddress, facebookId,
openId, locale, firstName,
        middleName, lastName, prefixId, suffixId, male,
birthdayMonth,
        birthdayDay, birthdayYear, jobTitle, groupIds,
organizationIds,
        roleIds, userGroupIds, sendEmail, serviceContext);
    UserLocalServiceUtil.updateAgreedToTermsOfUse(user.getUserId(), true)
```

- Portlets
 - Gekapselte Web Anwendungen als Teil des Portals
- Themes
 - Look And Feel des Portals
- Layout Templates
 - Seiten-Layout
- Hooks
 - Liferay ruft bei internen Aktionen Callback-Funktionen eines Hooks auf

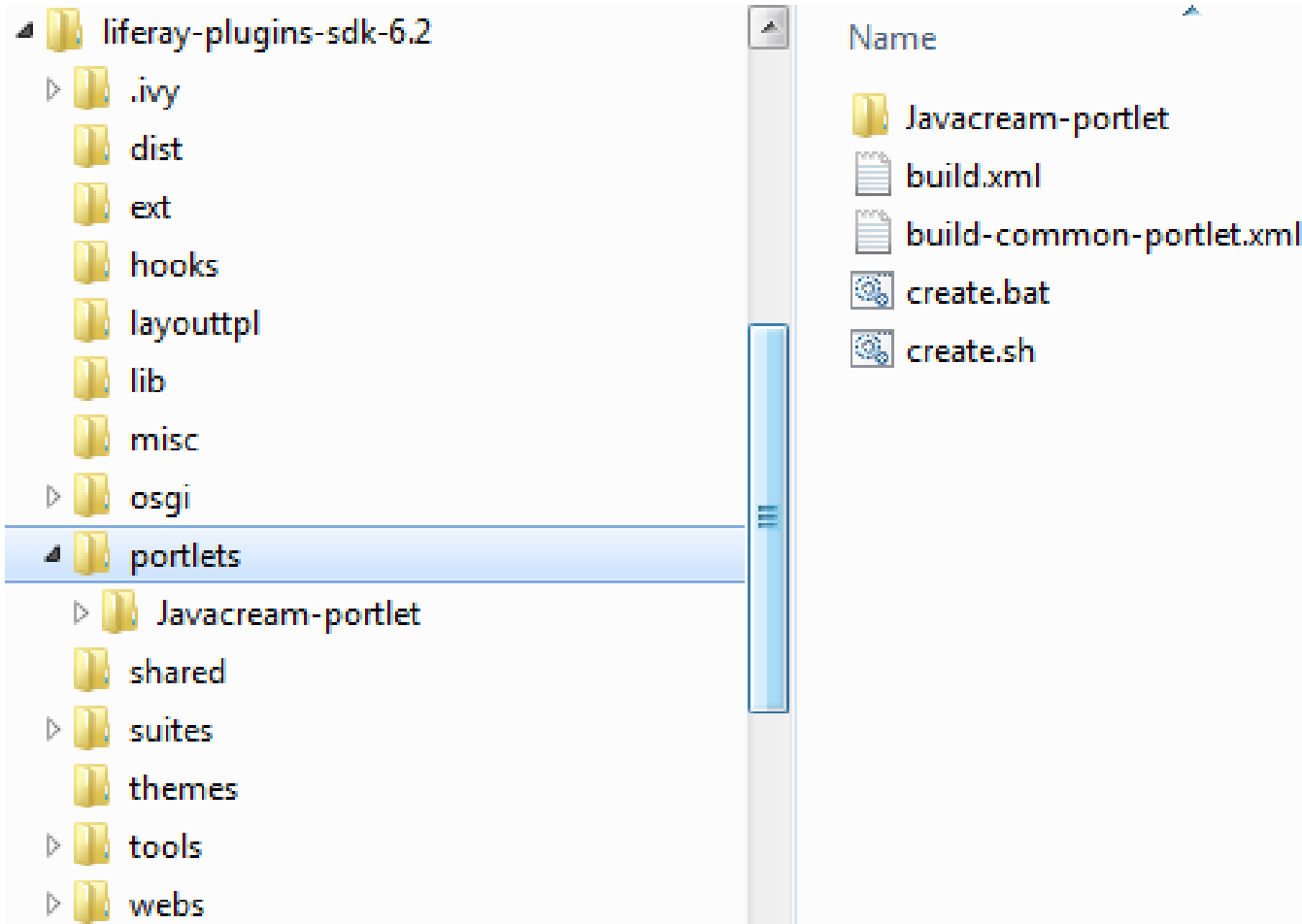
Webs

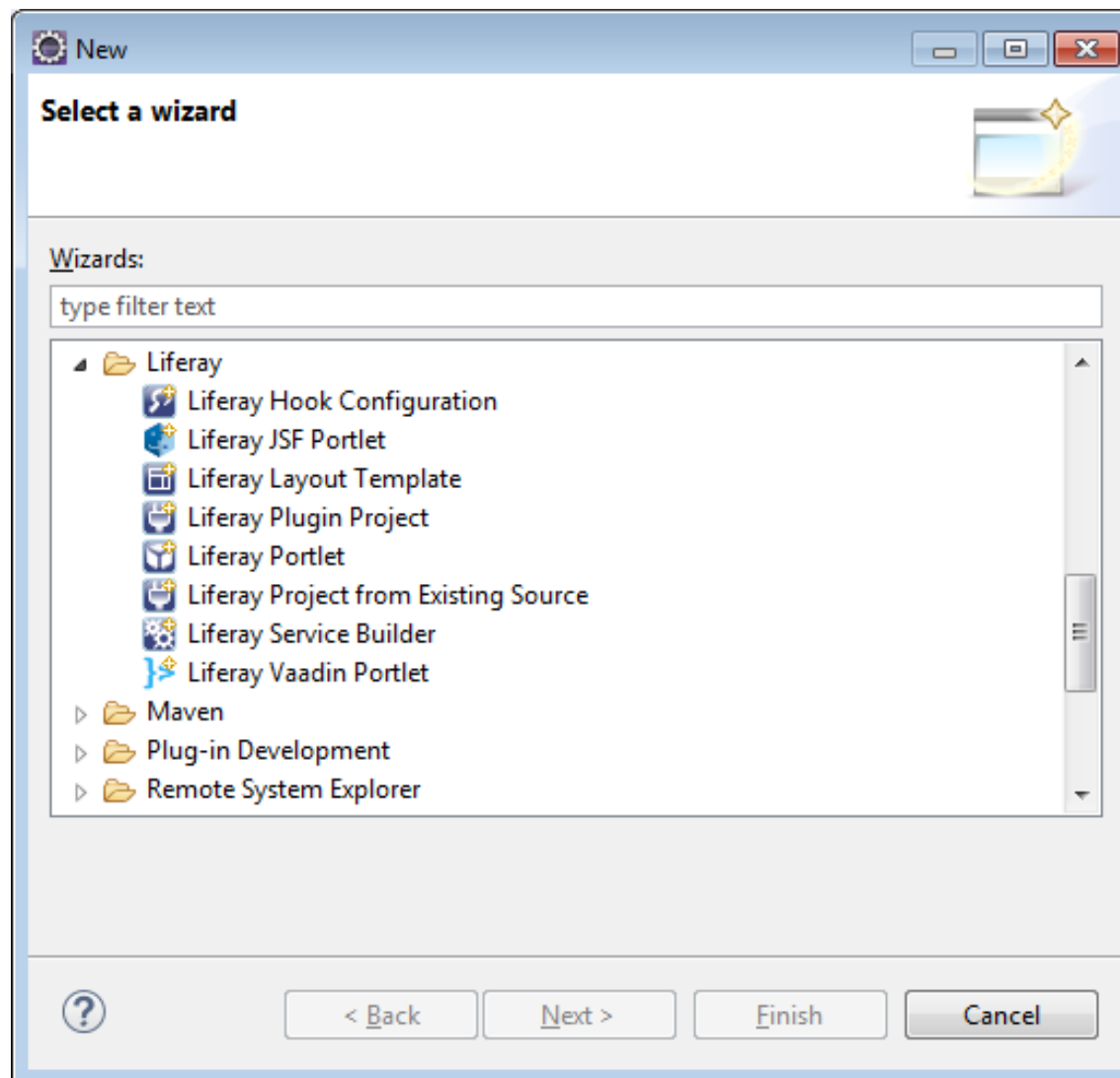
- Normale Web-Anwendungen, die jedoch Liferay-APIs und Services benutzen können
- Ext
 - Liferay kann als Open Source-Produkt direkt erweitert werden
 - Im Unterschied zu den anderen PlugIns ist kein Hot Deployment möglich!

3.2

LIFERAY SDK

- Skripte
 - Projekt-Generatoren
 - Bauen der Anwendung
 - Deployment in den Liferay-Server
- Ab Liferay 7: Blame CLI

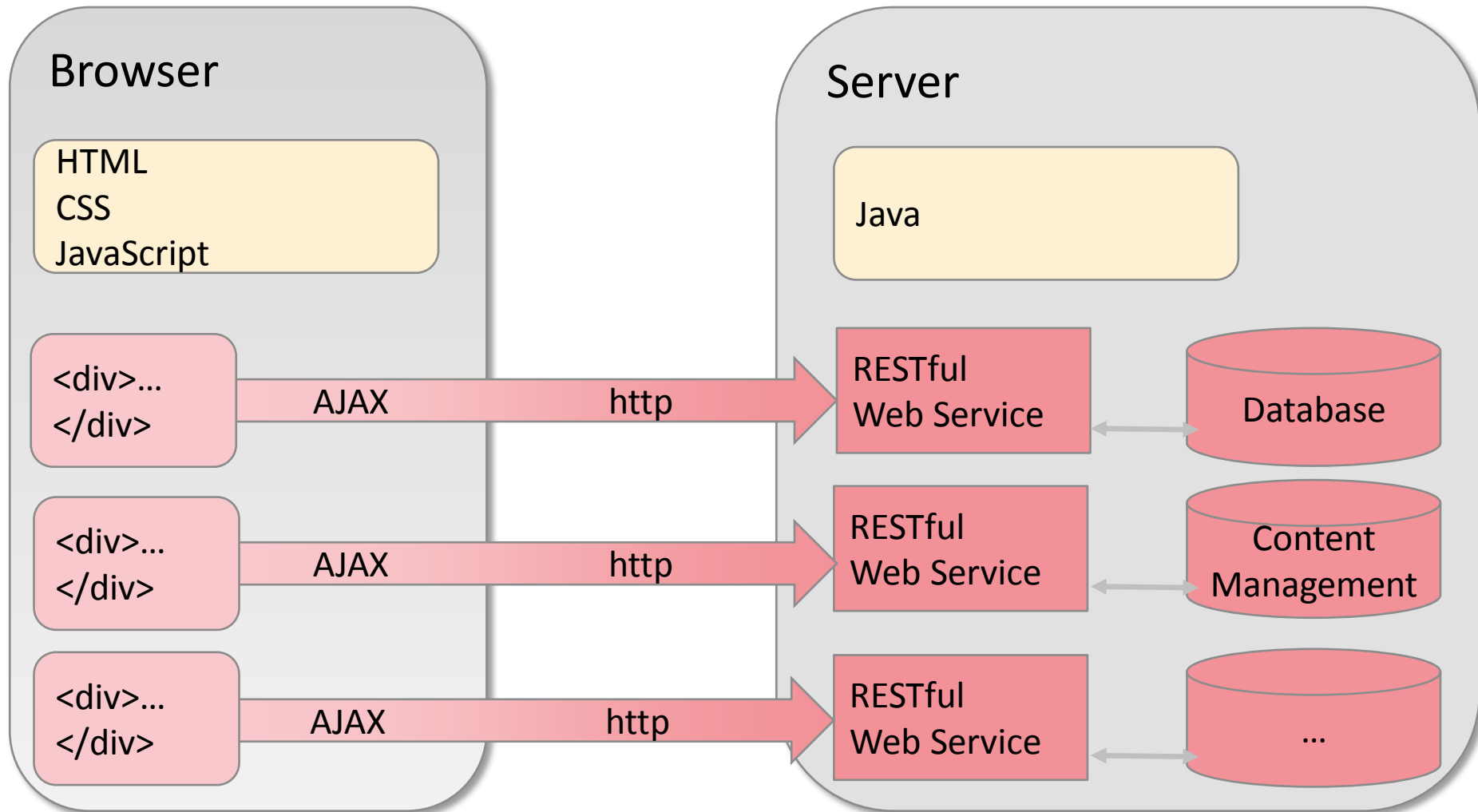


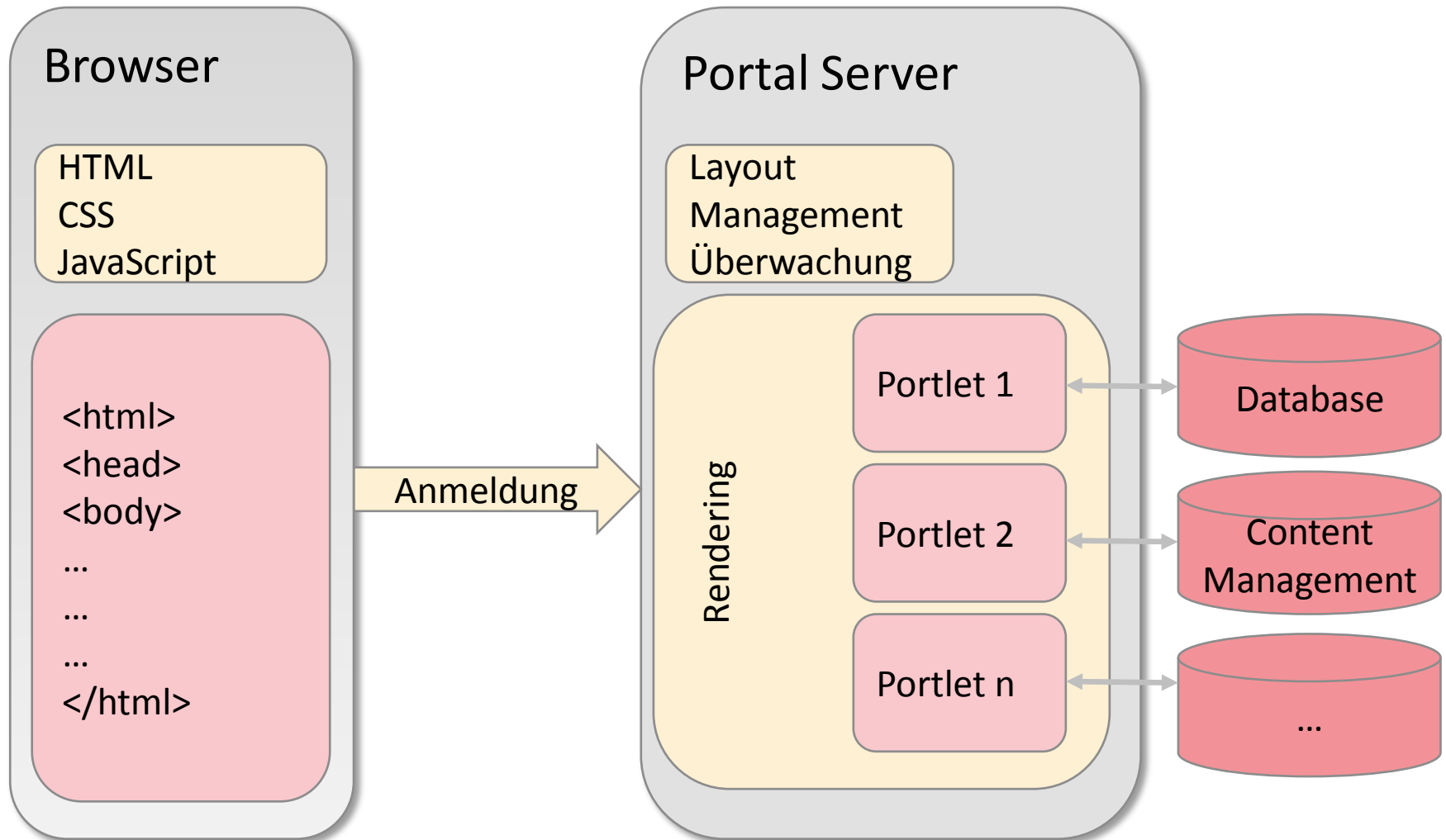


3.3

PORTLETS

- JSR 168 und JSR 286
- Darin sind enthalten
 - Programmiermodell
 - Laufzeit-Verhalten des Portal-Servers
 - Vergleichbar mit der Servlet-Spezifikation der JEE
- Ein Portlet ist eine Server-Seitige Komponente
 - Im Gegensatz zu JavaScript-basierten Client-Portlets
- Portlets müssen deshalb durch http-Requests mit dem Client-Zustand abgeglichen werden
 - Links und Formular-Submits
 - JavaScript-Events und Scheduling
 - AJAX-Technologie

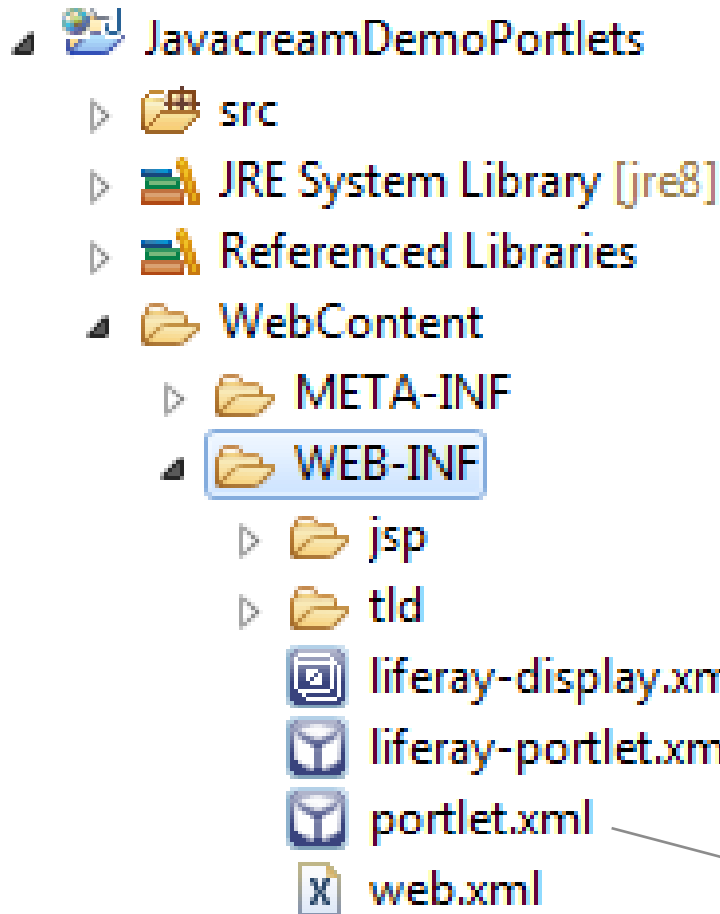




- Liferay-Apps sind technisch gesehen immer Portlets
- Allerdings unterscheiden sich die verwendeten Programmiermodelle gewaltig
 - "Simple" Portlets
 - Portlets mit JavaServer Faces und Portlet Bridge
 - Benutzung des Spring-Frameworks mit Spring MVC
 - Liferay-API

- Der Lebenszyklus eines Portlets wird vom Portal-Server komplett übernommen
- Der Portal-Server instanziert jedes Portlet nur eine einziges mal
 - Vorsicht: Multithreading-Effekte müssen damit beachtet werden!
 - Dieses Verhalten ist komplett analog zu einer Servlet-basierten Web-Anwendung
- Der Lebenszyklus eines Portlets wird in einem Interface definiert

- Meta-Daten
 - Deskriptive Elemente wie Icons, Beschreibung
- Angabe einer Java-Klasse
 - Der Lifecycle eines Portlets ruft Callback-Funktionen einer Java-Klasse auf
- Konfigurationsangaben für den Portal-Server
 - Wie oft pro Seite benutzbar?
 - Cache-Verhalten
 - Welche Request-Parameter werden zur Verfügung gestellt?
- Anbindung des Portlets an die Portal-Server-Infrastruktur
 - Event-Producer und Listener
 - Inter-Portlet-Kommunikation
- Zusätzlich noch Hersteller-spezifische Erweiterungen
 - Liferay



Liferay-Erweiterungen
für UI-Darstellung

Technische Liferay-
Erweiterungen

Standard-Portlet-
Deskriptor

- JEE-Applikationsserver können Anwendungen in der Regel "hot" bzw. automatisch installieren
 - Nützlich für Test und Entwicklung, weniger für den Produktions-Betrieb
- Problem:
 - Liferay ist selbst bereits die JEE-Anwendung
 - Die Installation eines neuen Portlets müsste damit zu einem Redeployment der Liferay-Anwendung führen
 - Das ist aus Zeitgründen nicht akzeptierbar
- Liferay-Lösung
 - Liferay definiert ein eigenes Hot-Deploy-Verfahren
 - Ein zweistufiger Prozess
 - Dabei wird ein Portlet als einfache WAR-Datei deployed
 - Vorsicht: Das Portlet ist ein WAR-Fragment und damit nicht alleine lauffähig
 - Anschließend wird das Web Archiv mit Liferay-spezifischen Erweiterungen versehen und als normale Anwendung vom Applikationsserver installiert

- Datenaustausch über ein gemeinsam genutztes Backend
- Theoretisch auch möglich: Java-Heap
 - Allerdings funktioniert das nur zwischen Portlets des gleichen Deployments
- Portlet-Session
 - Neben einer privaten Portlet-Session existiert auch ein Applikations-übergreifende Session
 - PORTLET_SCOPE
 - APPLICATION_SCOPE
 - Eine Portal-übergreifende Session ist in der Portlet-Spezifikation nicht vorgesehen
 - Wird aber von Liferay zur Verfügung gestellt
- Shared oder Public Render Parameters
 - Render-Parameter können für als `RenderRequest`-Parameter allen Portlets zur Verfügung gestellt werden
- Events
 - Event-Producer senden Events an ein Topic
 - Event-Listener registrieren sich an einem Topic

3.4

WEB PROGRAMMIERUNG

- Die Programmierung von Liferay-Anwendungen erfolgt primär auf Server-Seite mit Java
- Dabei werden Web-Anwendungen nach dem klassischen Request-Response-Prinzip konzipiert
 - Die Zuordnung der Requests zum Portlet der Anwendung sowie das optimierte Rendering übernimmt Liferay

- <https://GitHub.com/Javacream/org.javacream.training.liferay.jsf>

3.6

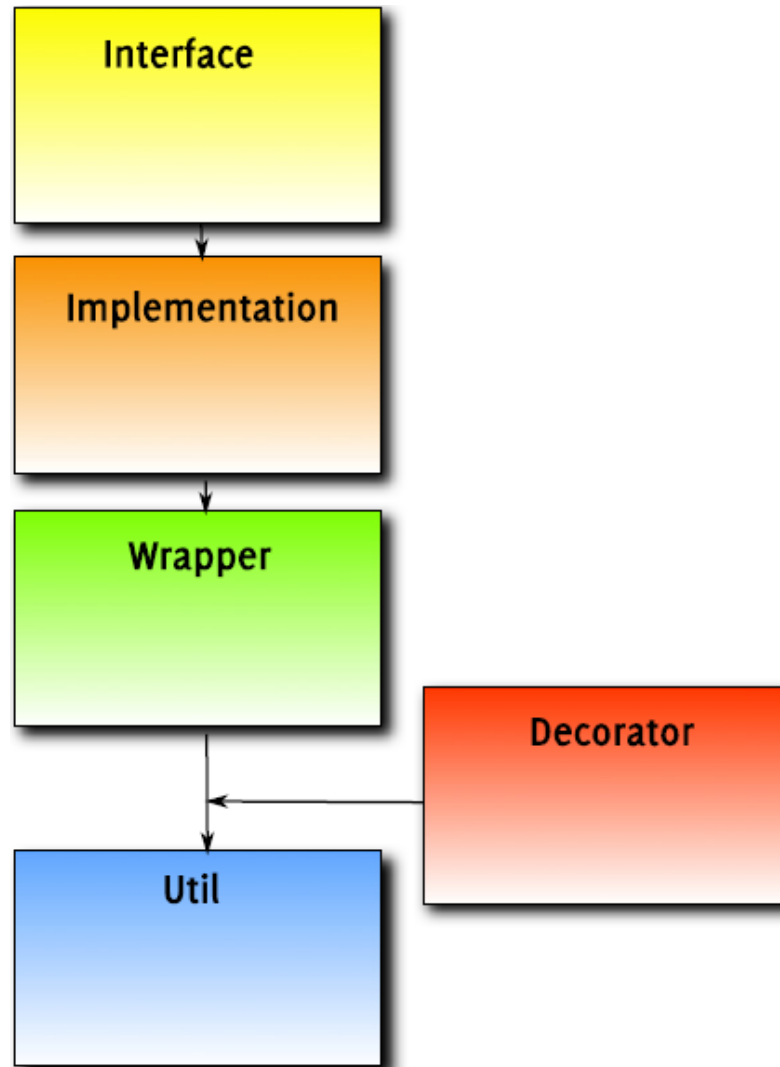
LIFERAY SERVICES

■ Liferay-Services

- Bieten den Zugriff auf die internen Dienste und Daten von Liferay
 - z.B. `com.liferay.portal.service.UserLocalService`
 - Erzeugung und Verwendung wird meistens noch über eine Utility-Klasse mit statischen Methoden vereinfacht

■ Eigene Services


- Werden nach Liferay-Richtlinien für Code definiert und implementiert
- Liferay-IDE stellt hierfür einen Wizard zur Verfügung
 - Service Builder



3.7

ÜBERWACHUNG

- Liferay ist ein Java-Programm
 - Damit sind die Metriken der Java Virtual Machine verfügbar
 - Speicher, insbesondere mit Garbage Collection
 - CPU und Threads
 - Auch der Applikationsserver stellt eigene Metriken zur Verfügung
 - Sessions
 - Zugriffs-Statistiken
- Zugriff
 - Bevorzugt über JMX
 - Liferay-Konsole

 Liferay

Control Panel

Users

Sites

Apps

Configuration

Virtual Instances

Instance Settings

System Settings

Sync Connector Admin

Components

Workflow Configuration

Custom Fields

Workflow Definition

Service Access Policy

Server Administration

TT Test

Server Administration

ResourcesLog LevelsPropertiesCAPTCHAData MigrationFile UploadsMailExternal ServicesScriptShutdown

Info: Liferay Community Edition Portal 7.0.6 GA7 (Wilberforce / Build 7006 / April 17, 2018) Uptime: 0:26:58

Used Memory / Total Memory

Used Memory / Maximum Memory

Used Memory 626,161,680 Bytes

Total Memory 975,175,680 Bytes

Maximum Memory 975,175,680 Bytes

System Actions

Run the garbage collector to free up memory.

Execute

Generate thread dump.

Execute

1.0 052018 © Javacream

Liferay

79

3.8

BENUTZERVERWALTUNG UND RECHTEKONZEPT

- Anbindung an externes Benutzerverwaltungssystem
 - Meistens ein Directory Server
 - Synchronisation über Export/Import
- Benutzer und Passwörter können auch über Liferay erfasst werden
 - Permission Policies

- Liferay definiert einen Satz von Standard-Rollen
 - Administrator
 - Guest
 - Site Administrator
 - ...
- Jedem dieser Rollen ist ein Satz von Permissions zugeordnet

- Praktische jeder Ressource in Liferay können fein-granulare Permissions zugeordnet werden
 - Sites
 - Pages
 - Portlets
 - Aktionen

Beispiel: Portlet-Permissions

Permissions							Search	Q
Role	Help	View	Preferences	Add to Page	Configuration	Permissions		
Guest	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
Portal Content Reviewer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Power User	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Site Content Reviewer	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
Site Member	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		
User	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>		