

Workshop

Softwarearchitektur

Vorgehen beim Workshop

- Architekturanalyse der einzelnen Beiträge
- Jeweils Bestimmung von:
 - Typen von Komponenten
 - Typen von Beziehungen (vor allem Interaktion)
- Ableiten von Architektureigenschaften/Mustern
- Herausarbeiten der Qualitätsmaße
- Herausarbeiten:
Welche Qualitätsmaße werden durch welche Architektureigenschaften/Muster unterstützt?

Ergebnis Architekturanalyse

Beispiel	Komponenten	Beziehungen
MDS	Endgeräte (heterogen), Proxy (MDS) Server (LMS)	via Proxy, lose gekoppelt, transformierend
SBK	SW-Objekt, Konfigurations-Objekt (Interface)	erzeugend bzw. als Input (Entwurfsmuster)
Versions- Manager	SW-Komponenten (allgemein)	Konfiguration dynamisch zur Laufzeit semantisch beschrieben
Web-Service Komposition	Web Services, semantische Brücken	semantisch beschrieben (via Ontologie)
ALV	a) SW-Komponenten in Schichten b) Applikationen, Stubs	a) von oben nach unten
OMQ	a) SW-Komponenten (allgemein) b) Clients, Kernel, Services	a) via nachrichtenorientierte Middleware
Kollaboratives OSGi	OSGs, Bundles, Services, Bundle-Proxies, Service-Proxies	transparente Verteilung
ASCML	(normale) Agenten, ASCML (Hebamme, Big-Brother)	zentrale Konfiguration

Ergebnis Muster vs. Qualitätsmaße

	Entwicklungskosten	Wartungskosten	Langlebigkeit	Performanz/Skalierbarkeit	Maintainability/Wartbarkeit	Zuverlässigkeit (Fehlervermeidung)	Erweiterbarkeit	Sicherheit	Portabilität (Plattformunabhängigkeit)	Benutzerfreundlichkeit	Anpassbarkeit/Flexibilität	Fehlertoleranz	Korrektheit	Entwicklerfreundlichkeit	Wiederverwendbarkeit	Interoperabilität	Beispielprojekte
Lose Kopplung	-	+	+		+		+				+	+					MDS
Proxymuster				+												+	MDS
Mehrschichten	0	+			+		+				+						ALV
MVC (Design Patterns)					+		+				+			+	+		LMS
Middleware	+			+					+		+			+		+	OMQ
Kapselung		+	+			+					+			+		+	OSGI
Struktur (zB. Baum)	+	+	+			+	+							+			Versionsmanager
geeignete Wahl von Fremdkomponenten	+	+															ALV
Standards			+			+			+					+	+	+	ASCML
Caching				+													ALV
Kompression				+													MDS / OSGI
Einsatz moderner Programmiersprachen (z. B.GC)	+	+											+	+			ALV
Constraints										+		+	+				Versionsmanager
Dependency-Graph/Relationenservice	-										+	+		-			personalisierte Dienste
Abstraktion							+		+						+	+	alle
Virtual Machines (zB. Java)									+		+						MDS / Versionsmanager
Einfachheit		+			+	+								+			ALV
Modularisierung		+			+		+				+				+		Versionsmanager