

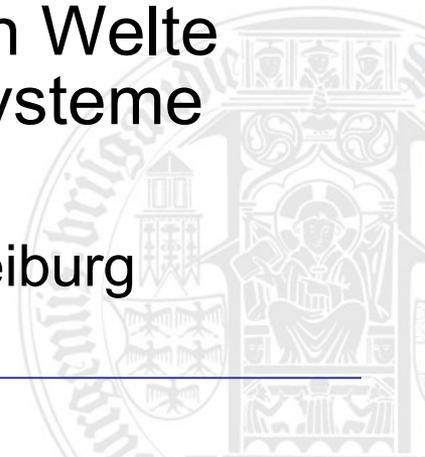


Kunstgewerbemuseum in Berlin Workshop der *nestor* Media AG 14.03.2008

Emulation – Wiederherstellung von Ablaufumgebungen Erfolgsbedingungen von Emulation als Langzeitarchivierungsstrategie

Dirk von Suchodoletz / Randolph Welte
Lehrstuhl für Kommunikationssysteme
Prof. Schneider

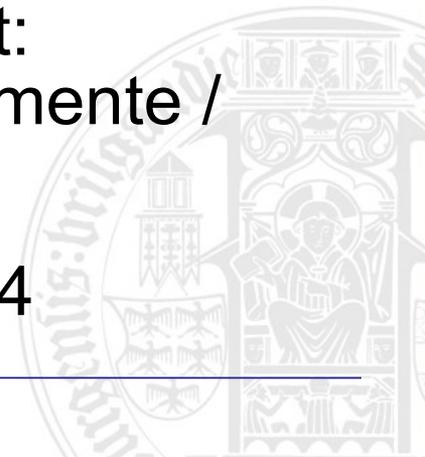
Rechenzentrum der Universität Freiburg





Lehrstuhl für Kommunikationssysteme I

- Randolph Welte - rwelte@rz.uni-freiburg.de
 - Mitarbeiter am Lehrstuhl für Kommunikationssysteme / Rechenzentrum
 - Applikations- und Webserviceentwicklung
 - Realisierung und Entwicklung des “Web Transport Tools” - Promotionsprojekt: “Langzeitarchivierung digitaler Dokumente / Webtransport”
 - Sozialisiert mit ZX Spectrum und C64
-





Lehrstuhl für Kommunikationssysteme II

- Dirk v. Suchodoletz - dsuchod@rz.uni-freiburg.de
 - Assistent am Lehrstuhl
 - Entwicklung und Betrieb der RZ-Lehrpools:
Stateless Linux mit bspw. WinXP in VM-Player
 - Erster Computer: ZX Spectrum, ab 1992 PC
 - Dadurch generelle Beschäftigung mit
Virtualisierungslösungen und Emulation
-





Emulation als generelle LA-Strategie I

- Emulation nicht nur auf “einfache” Architekturen, wie Home-Computer, Arcade, Konsolen beschränkt
 - Generelles Konzept auch für aktuelle, zukünftige Architekturen
 - Jedoch mit technologischen Weiterentwicklung zunehmende Komplexität
 - Zusätzlich einige Rand- und Erfolgsbedingungen zu berücksichtigen
-





Emulation als generelle LA-Strategie II

- Betrachtung bereits existierender Ansätze und Programme in Bezug auf bisherige Architekturen
 - Dynamische digitale Objekte der PC-Ära
 - Anforderungen an Objektzugriff und damit beauftragte Institutionen
 - Konzepte der Umgebungswiederherstellung
 - Softwarearchiv aus Sekundärobjekten
-





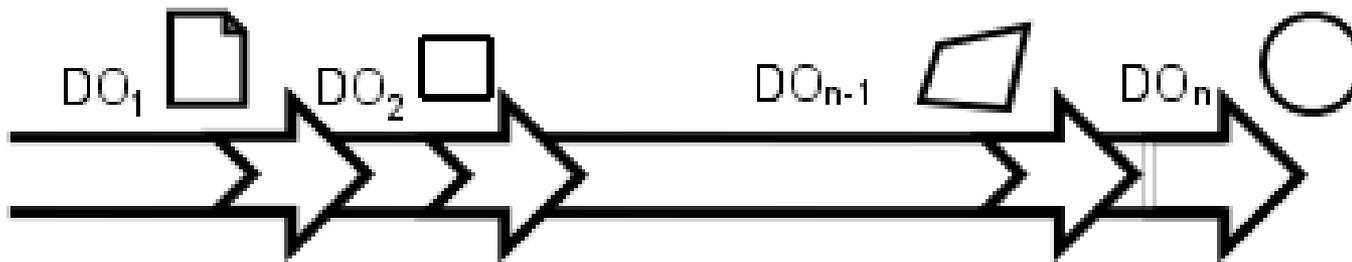
Emulation als generelle LA-Strategie III

- Forschungen an der Universität Freiburg im Rahmen des EU-geförderten PLANETS Project
 - Zusammenschluss von europäischen Bibliotheken, Archiven, Forschungseinrichtungen und Technologiefirmen



Klassische Langzeitstrategie – Migration

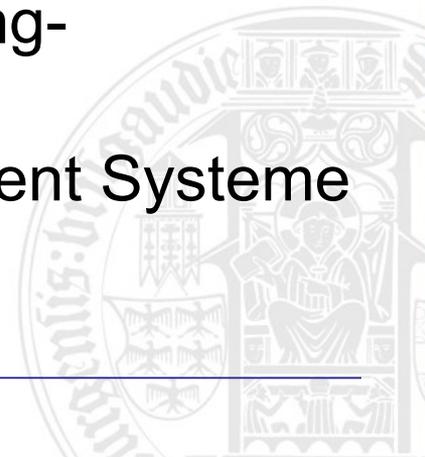
- Anwendbar auf klassische Archivalien
 - Dokumente, Audio, Video, ...
 - Statisch – keine User-Interaktion
 - Migrationsvorgänge gut automatisierbar, jedoch nicht ganz unproblematisch – Authentizitätsproblem des n-fach migrierten Objekts





Dynamische digitale Objekte I

- Darüber hinaus große Anzahl *dynamischer* digitaler Objekte
 - Objekte benötigen passende Laufzeitumgebung
etliche davon *interaktiv* (Benutzerinteraktion)
 - Computerprogramme jedwelcher Art
 - Betriebssysteme
 - Digitale Enzyklopädien, E-Learning-Umgebungen
 - Datenbanken, Content Management Systeme
 - Multimedia-Applikationen, Spiele



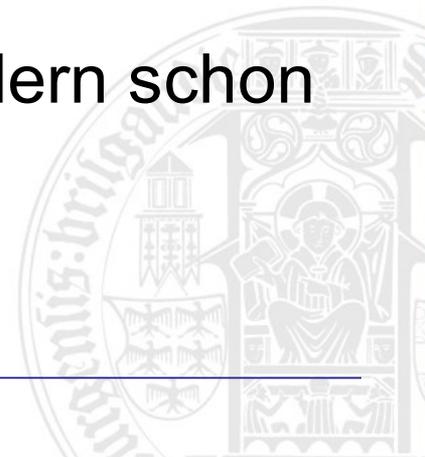
Dynamische digitale Objekte II





Archivierungsstrategie – Emulation

- **Emulation**
 - Setzt nicht am Objekt selbst an
 - Stattdessen wird notwendige Ablaufumgebung des Objekts erhalten
 - Vorteil: Objekt bleibt unverändert
 - Auch Ergänzungsstrategie zur Migration
- Emulation kein neues Konzept, sondern schon länger in anderen Bereichen genutzt





Emulation / Virtualisierung

- Beispiele DOS-Emulation im Windows NT, Apple - "Classic" als Emulationsumgebung älterer Mac-OS 8/9 Applikationen für Mac-OS X
 - Strategie des Softwareübergangs zwischen Betriebssystemen und Architekturen
 - Sehr verwandt, trotzdem sauber abzugrenzen:
Virtualisierung
 - Elemente von Emulation – Peripherie
 - Jedoch direkte CPU-Nutzung
-





Klassen von Hardware I

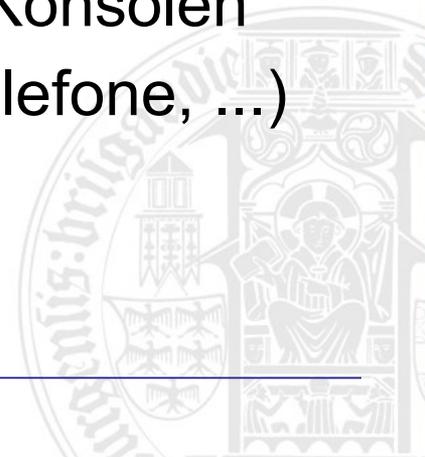
- Grobe zeitliche Einordnung, andere Kategorisierungen denkbar
 - Großrechner und Mainframes





Klassen von Hardware II

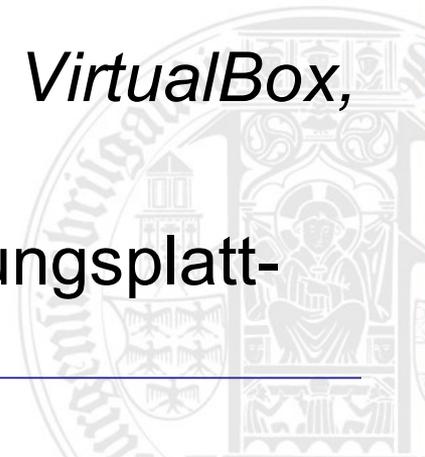
- Home-Computer, Arcade, Videospiele im vorherigen Vortrag bereits erörtert
- Verschiedene Apple Systeme, Unix Workstations
- X86 vom 8bit 8086 bis zu 64bit AMD und Intel CPUs
- Diverse neuere und neueste Game-Konsolen
- Eingebettete Systeme (PDA, Mobiltelefone, ...)
- Home-Entertainment ...





Emulatoren – Beispiele I

- Mainframes – Computer History Simulation Project, Hercules
 - Home-Computer, Arcade – MESS/MAME, Hatari, UAE, Jasper, JaC64, AranyM, Arnold, PCSX, GNUBOY, ...
 - Apple – Basilisk (II), Pear PC, QEMU, Mini vMAC
 - X86 – QEMU, Bochs, Dioscuri, JPC, *VirtualBox*, *VMware*, *Virtual PC*, *Parallels*, ...
 - PDA, Embedded – diverse Entwicklungsplattformen der Hersteller, Pose, ...
-



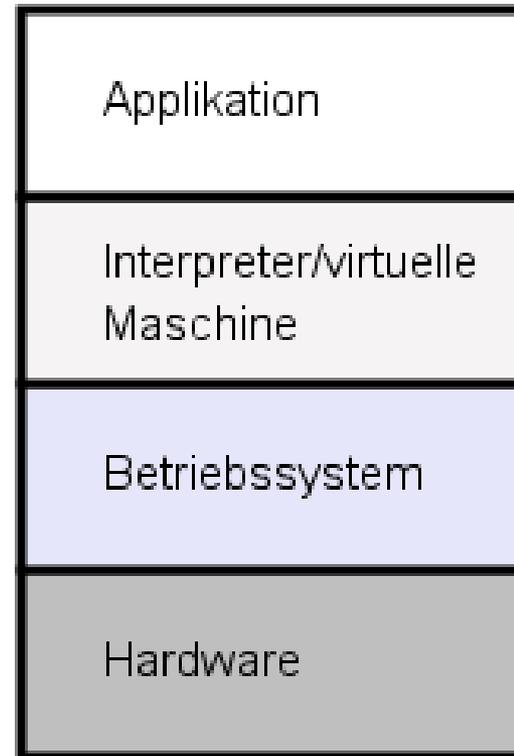


Emulatoren – Beispiele II

- Dosemu, Wine, ReactOS, ...
 - ScummVm als Interpreter einer Reihe von Lucas Arts und Sega Spielen für aktuelle Umgebungen
 - ...
 - Unterscheidung, Abgrenzung, Strukturierung vorzunehmen
 - In welcher Umgebung lauffähig (Betriebssystem, Rechnerarchitektur)
 - Welche Anforderungen an Ein- und Ausgabe
-

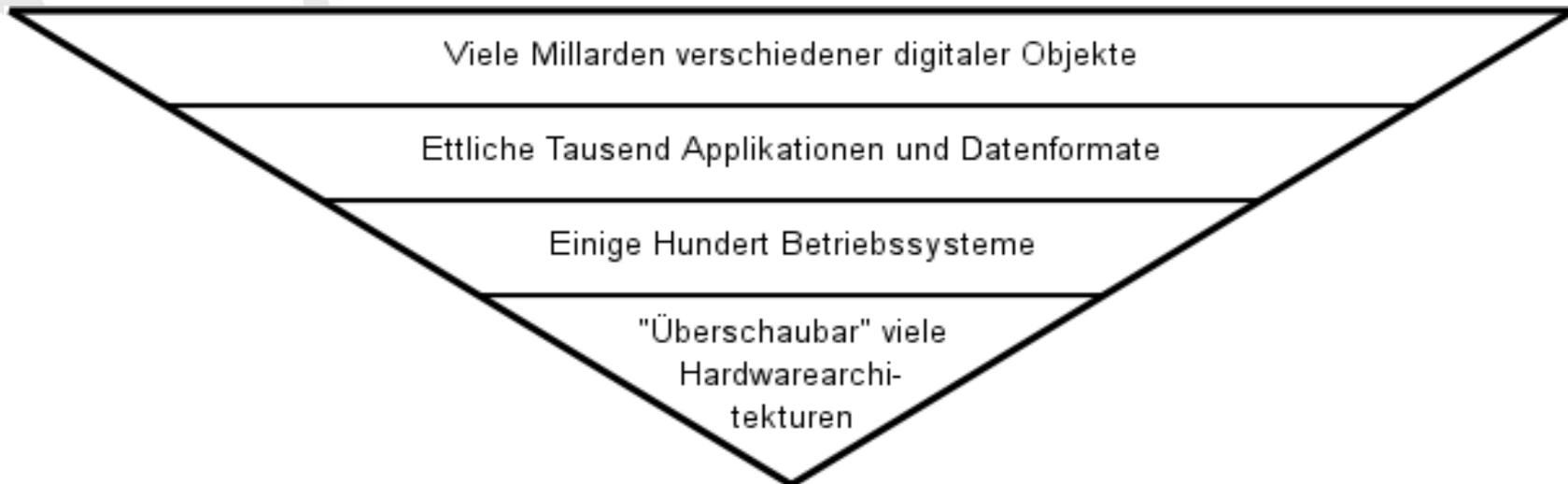
Emulation: Ansatzpunkte I

- Verschiedene Ebenen im Software/Hardware Stack
 - Applikation
 - Interpreter, Skriptsprachen (Perl, Python, Ruby, ...), JVM usw.
 - Betriebssystem
 - Hardware



Emulation: Ansatzpunkte II

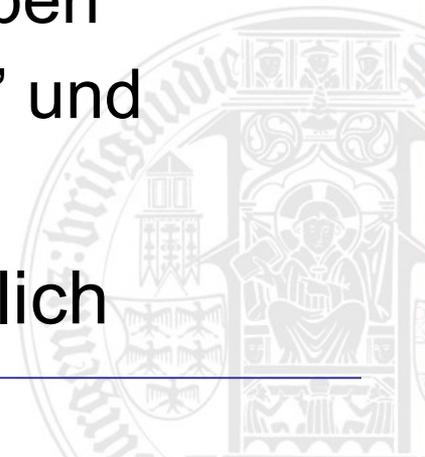
- Überlegung – Anzahl der notwendigen Emulatoren
- Ebene der Hardware attraktiv





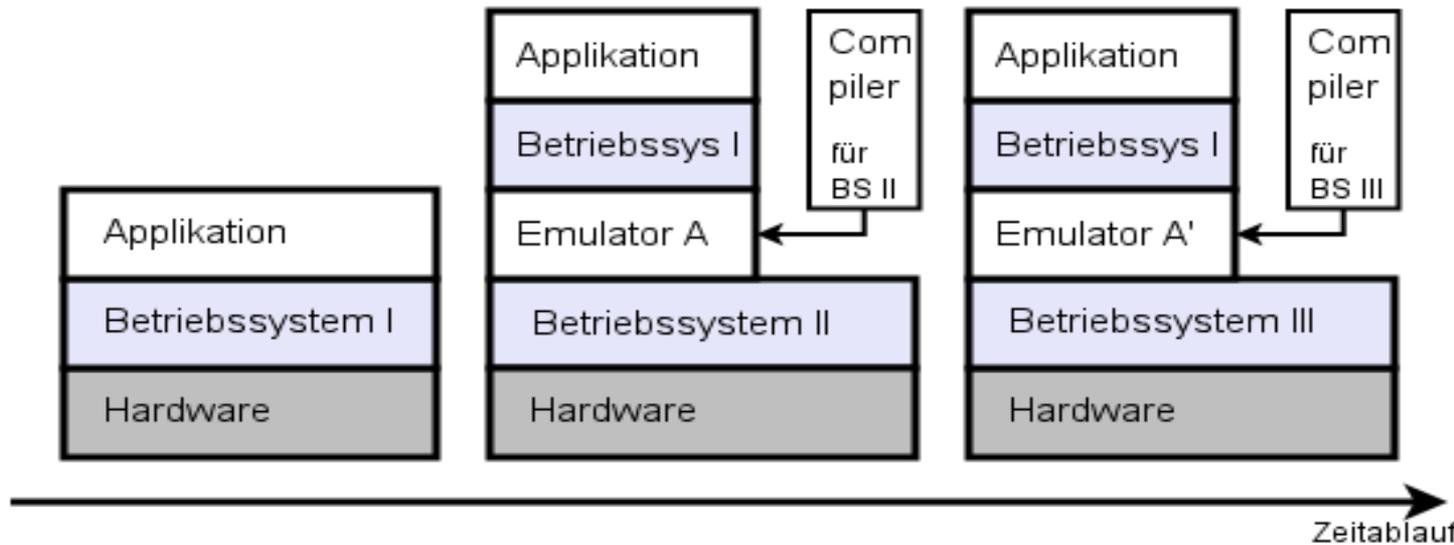
Emulation – Herausforderungen

- Emulatoren – Brückenfunktion zwischen alter und aktueller Umgebung
 - Selbst wieder Software
 - Aktuellen Anforderungen an ihre Umgebung als Applikation auf OS/Hardware
 - Übersetzung von Ein- und Ausgaben
 - Datenaustausch zwischen “innen” und “außen”
 - Regelmäßige Updates unumgänglich
-



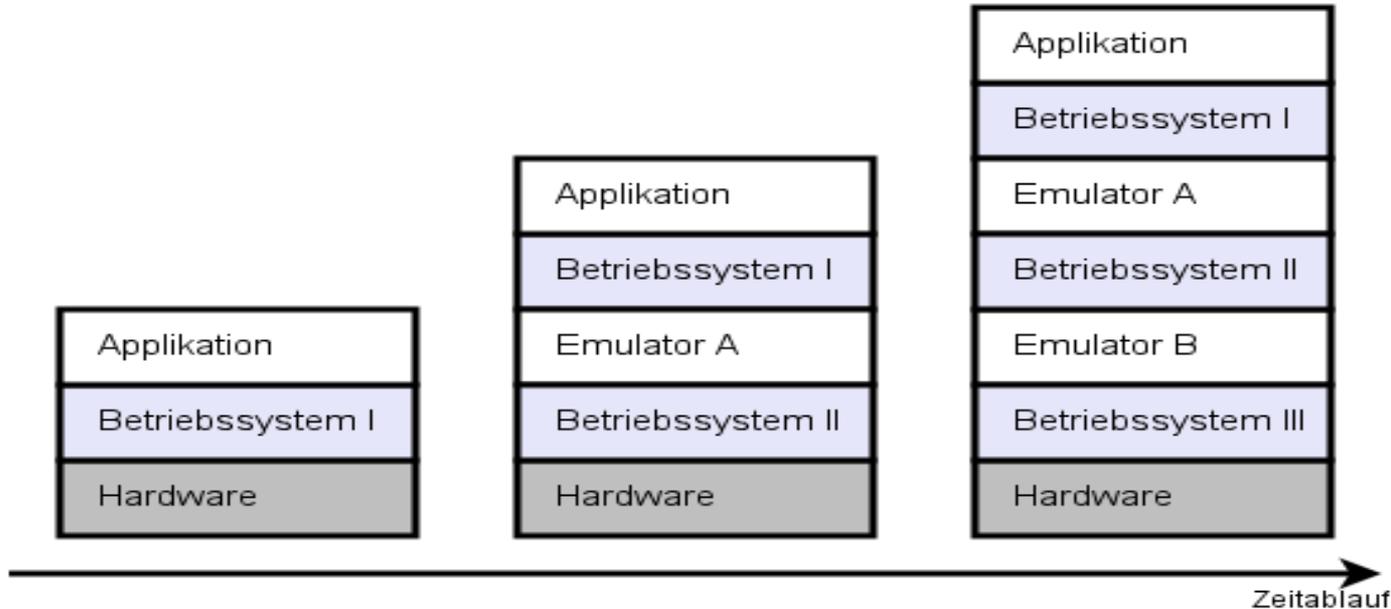
Emulatoren – Langzeitverfügbarkeit I

- Migration – Mitführen des Emulators mit der technischen Entwicklung
 - Anpassung der äußeren Schnittstellen



Emulatoren – Langzeitverfügbarkeit II

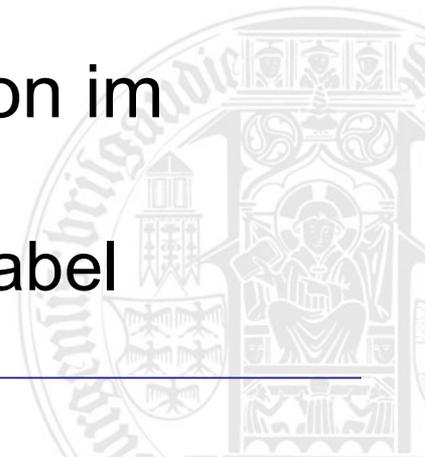
- Schachtelung, “Stacking” – gesamte Umgebung des alten Emulators abbilden
 - Möglicherweise Transport- und IO-Probleme





Emulatoren – Langzeitverfügbarkeit III

- Universal Virtual Machine (UVC/UVM)
 - Abstraktes Komponentensystem
 - Fixe Module für Emulation einzelner Hardware-komponenten
 - Beschreibungssprache zur Kombination der Module zu jeweiligem Plattformemulator
 - Modulare Emulationsansätze schon im Einsatz/Entwicklung
 - Beispiele: Dioscuri, QEMU, IronBabel
-





Emulation – Herausforderungen

- Keine simple Antwort auf alle Fragen
 - Emulator alleine reicht nicht, zusätzlich *digitale Sekundärobjekte*
 - Externe, interne Werkzeuge (Hilfsprogramme)
 - Metadaten zur Beschreibung von Emulator und benötigten weiteren Objekten
 - Fonts, Codecs, ...
 - Anleitungen, Handbücher
-

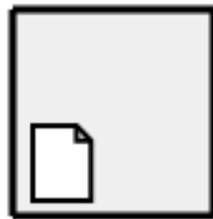


Bestimmung der Werkzeuganforderungen

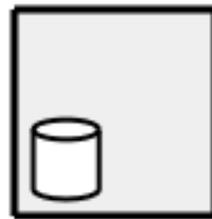
- *View-Path* als Modell zur Ermittlung notwendiger Sekundärobjekte
 - Primärobjekt kann Applikation erfordern
 - Applikation benötigt Ablaufumgebung in Form eines Betriebssystems (typischerweise)
 - Betriebssystem erfordert Hardwareumgebung



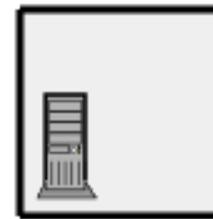
Digitales
Objekt



Datei-
Betrachter



Betriebs-
system



Rechner-
Architektur

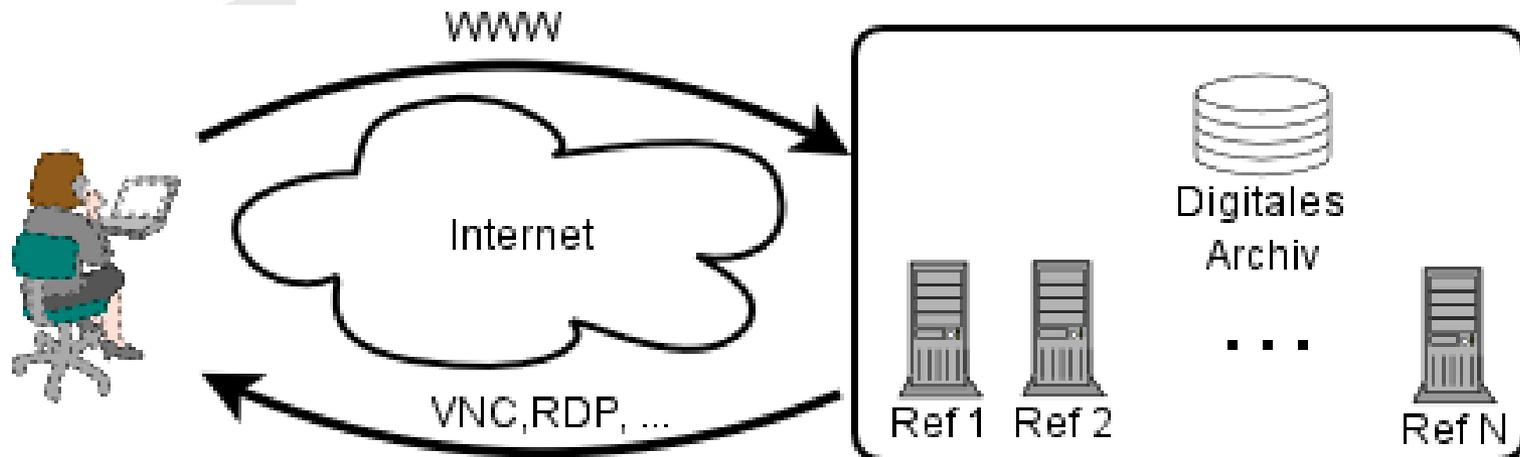


Referenzumgebungen I

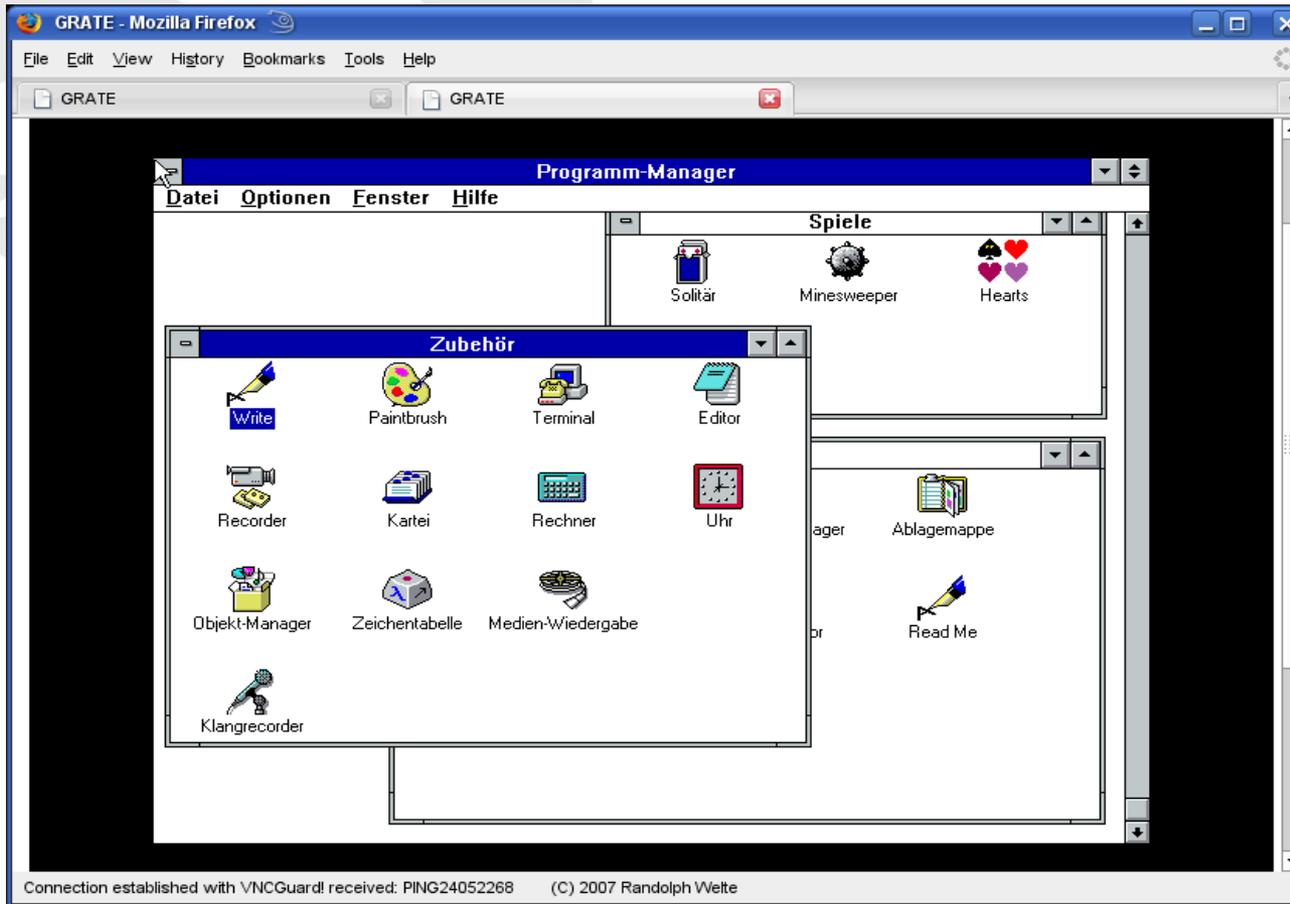
- Endpunkt des jeweiligen View-Path, damit zweiter Bezugspunkt für Emulator
 - Softwareumgebung auf die Emulatoren beständig angepasst/aktualisiert werden müssen
 - Umgebung mit der Endanwender arbeitet
 - Veränderlich mit technologischem Fortschritt
 - Archivmanagement hält sowohl Softwarearchiv als auch Referenzumgebungen im Blick
-

Referenzumgebungen II

- Verschiedene Varianten denkbar
 - Linux-/Windows-Workstations
 - Server für Web-Remote-Services
 - Container für virtualisierte Umgebungen



Referenzumgebungen III





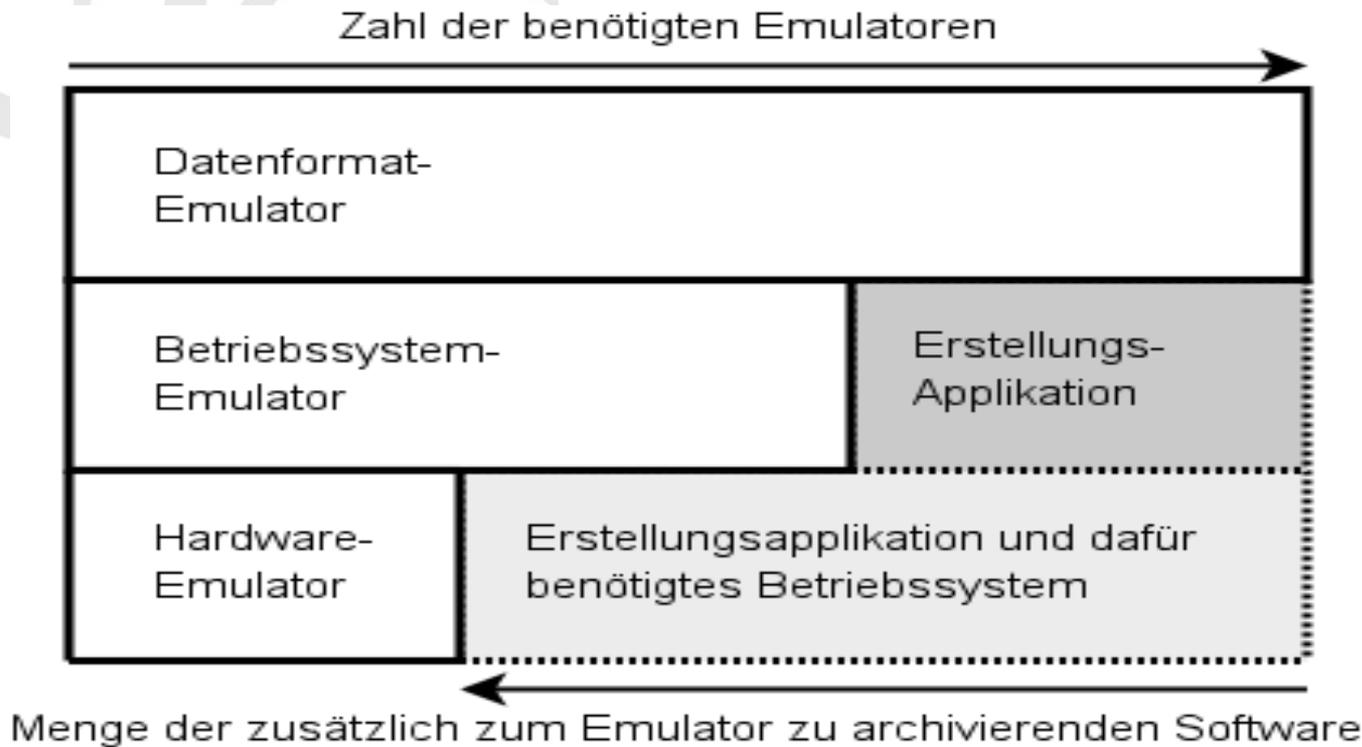
Emulation – Softwarearchiv I

- View-Path durch Objekt und Referenzumgebung bestimmt
 - Mehrere View-Path denkbar
 - Verschiedene Applikationen können ein Objekt (unterschiedlich gut) rendern
 - Applikationen können auf unterschiedlichen Betriebssystemen laufen
 - Alte Nutzungsumgebungen können in verschiedenen Emulatoren arbeiten
-



Emulation – Softwarearchiv II

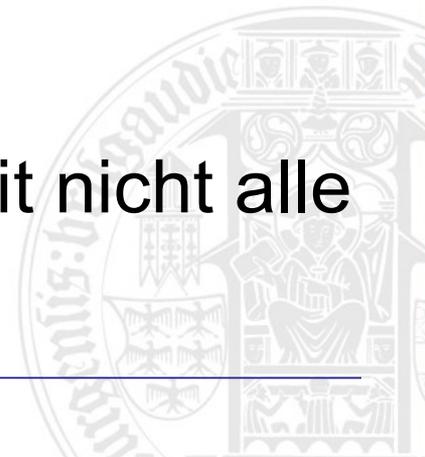
- Werkzeug-Mengen-Matrix





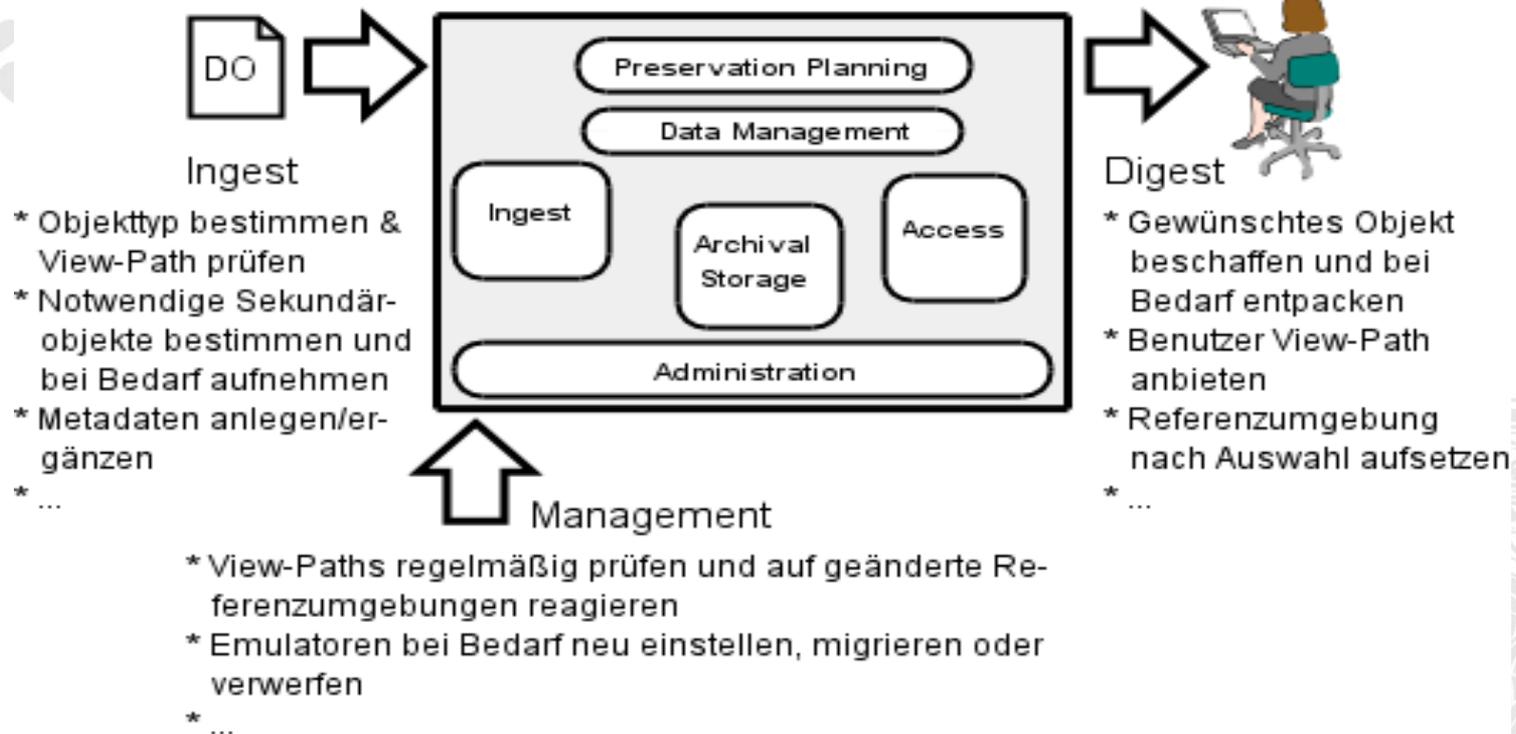
Emulation – Softwarearchiv III

- Softwarearchiv notwendiger Sekundärobjekte
 - Alle notwendigen Applikationen
 - Fonts, Codecs, Plugins, ...
 - Hilfsprogramme, wie Packer der verschiedenen Typen
 - Betriebssysteme
 - Treiber
 - View-Path zusammenfassbar – damit nicht alle Varianten von allem zwingend
-



Emulation – Archivmanagement I

- Aufgaben des Archivmanagements





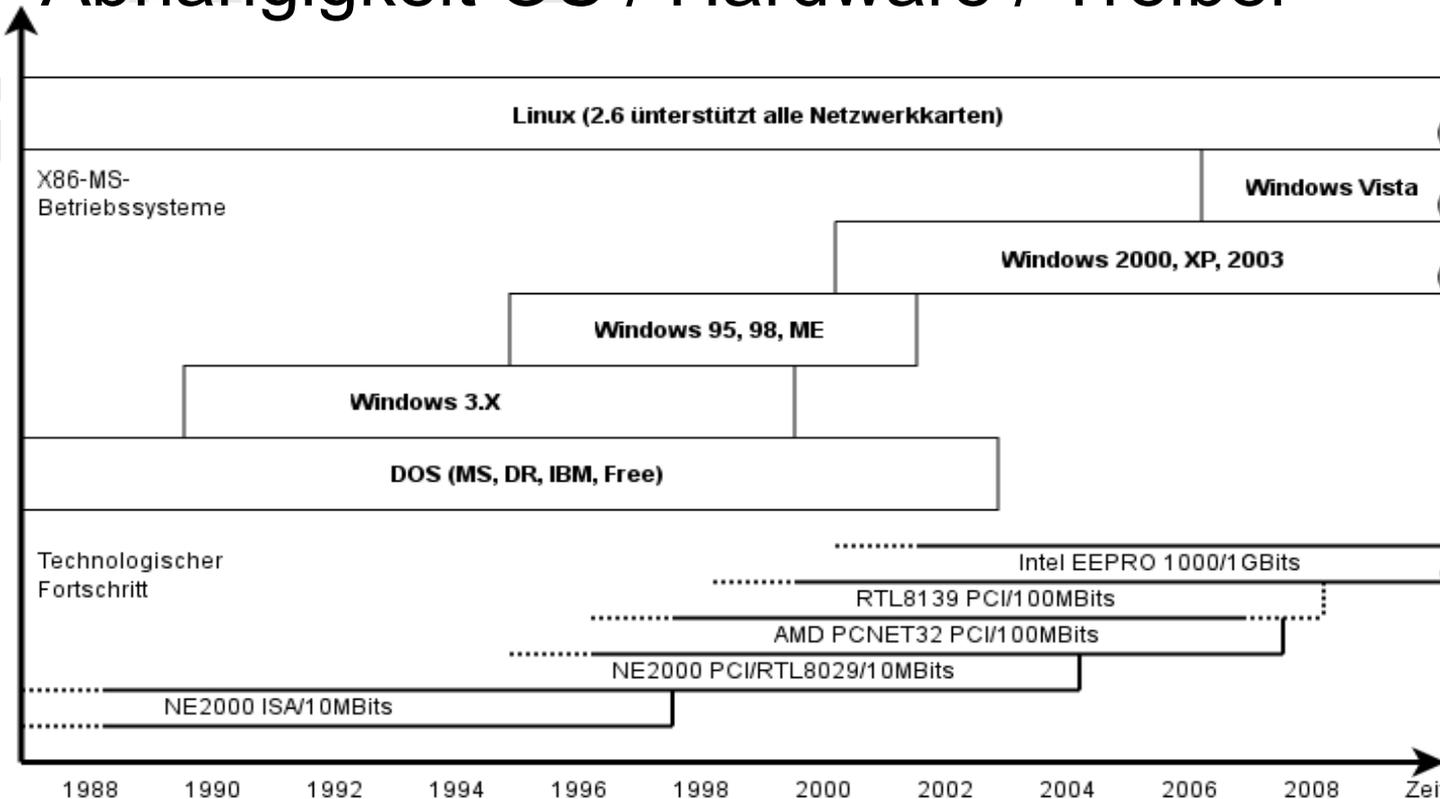
Emulation – Archivmanagement I

- Aufgaben des Archivmanagements
 - Objekte beim Archiveingang prüfen und um notwendige Metadaten für Emulation ergänzen
 - Regelmäßige Überprüfung aller View-Path für enthaltene Objekte
 - Pflege eines Archivs von Sekundärobjekten
 - Vereinfachungen – View-Paths aggregierbar
 - Bei Wechsel der Referenzumgebung – Anpassung der Emulatoren notwendig
-



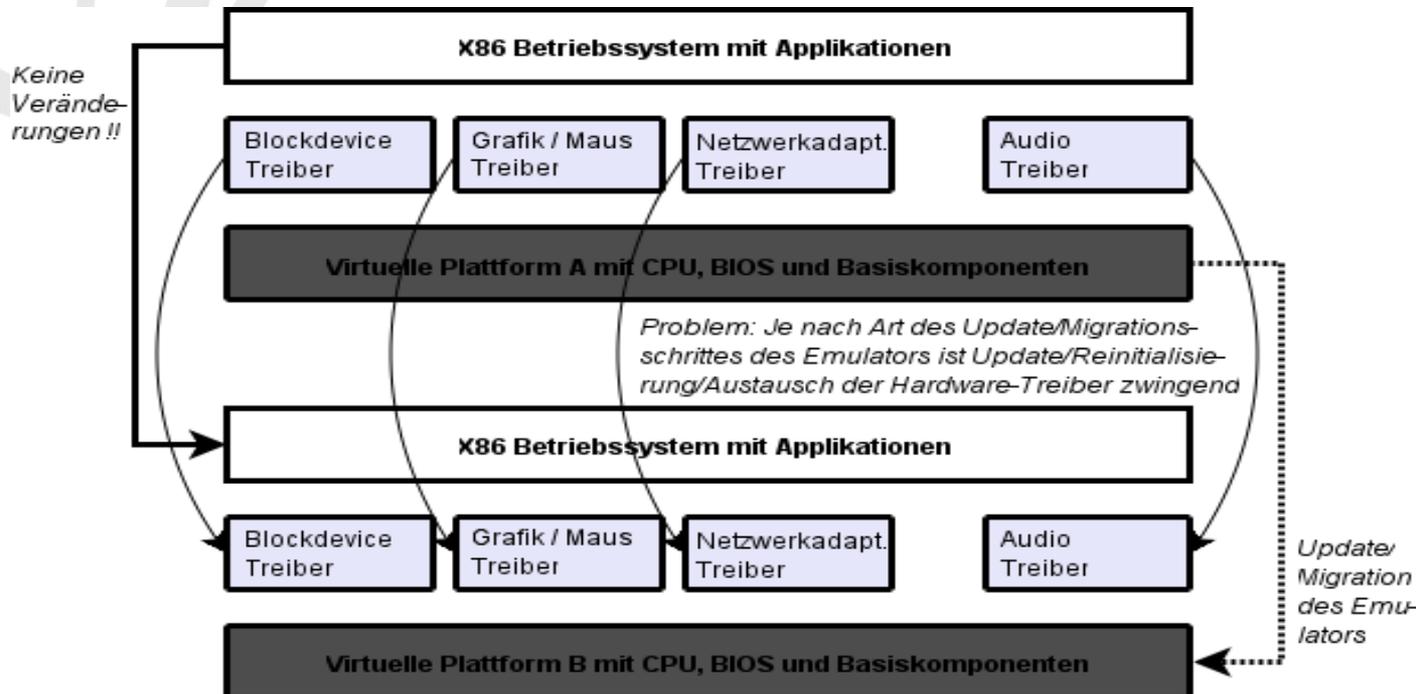
Emulation – Archivmanagement III

- Abhängigkeit OS / Hardware / Treiber



Emulation – Archivmanagement III

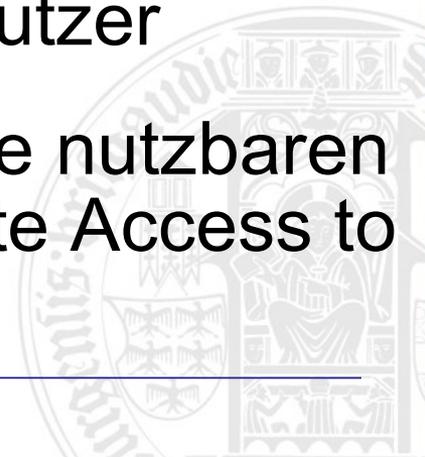
- Update-Problem emulierter Umgebungen





Emulation – Anwendung heute

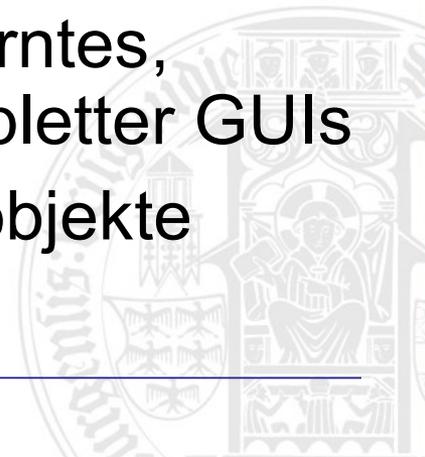
- Beispiel der verschiedenen Virtualisierer – hier VMware Workstation/Player
 - Beispiel QEMU als X86/PowerPC/Sparc ... Emulator
 - Problem der Installation etlicher Komponenten auf den Geräten der Endnutzer
 - Hierzu: Realisierung eines Remote nutzbaren Services – GRATE (Global Remote Access to Emulation-Services)
-





Emulation – zukünftige Anwendung I

- GRATE
 - Entwicklung am Lehrstuhl – “Fernsteuerung von Emulationsservices”
 - Plattformunabhängig in jedem Java fähigen Webbrowser ausführbar (Java /TightVNC basiert)
 - Via Application Publishing – entferntes, ortsunabhängiges Benutzen kompletter GUIs
 - Upload “eigener” digitaler Primärobjekte (Dateien)
-





Emulation – zukünftige Anwendung II

- GRATE
 - Identifizierung durch Format-Identifizierer und -Registries wie PRONOM / DROID
 - Bereitstellung und Nutzung hierfür geeigneter “Caches” emulierter Ablaufumgebungen
 - Download modifizierter Primärobjekte auf das lokale System des Benutzers
 - Übertragung von bestimmten “Keystrokes” wie CTRL-ALT-DELETE zum emulierten System (üblicherweise vom darunterliegenden System “abgefangen”)
-



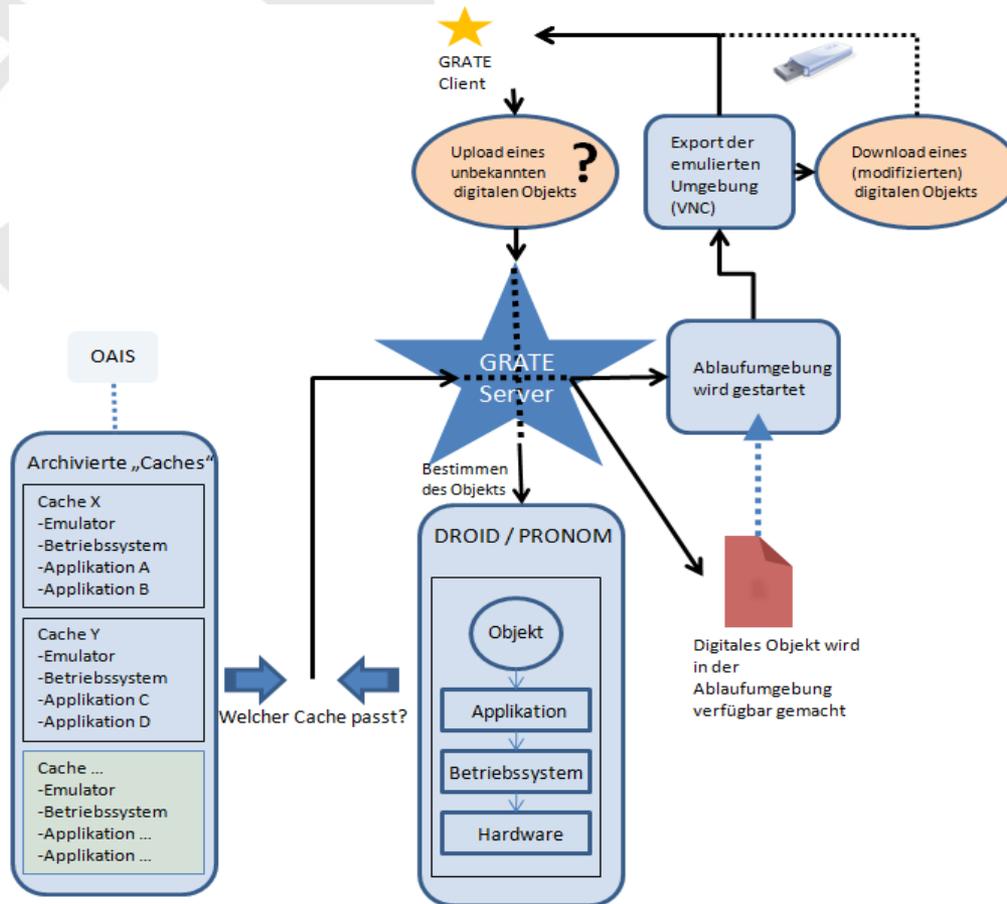


Emulation – zukünftige Anwendung III

- Vorteile:
 - Auflösung des Lokalisationsproblems lokaler Reference-Workstations – so deutlich breitere Benutzerschicht erreichbar
 - Lizenzmanagement (DRM): Provider soll komplette Kontrolle über seine Lizenzen behalten
 - Keine Softwareinstallation auf der Benutzerseite erforderlich, nutzbar mit jedem Webbrowser
 - Zentrale Administration – Kostenreduktion
-



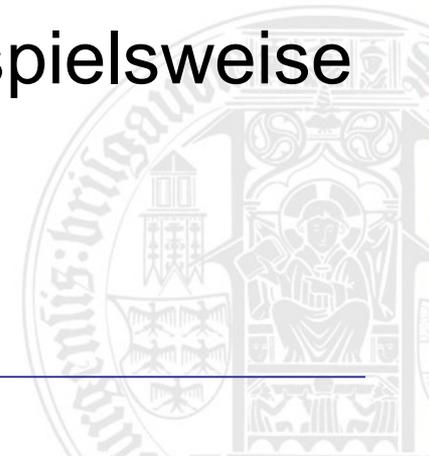
GRATE – Übersicht





Remote-Emulation – Offene Punkte

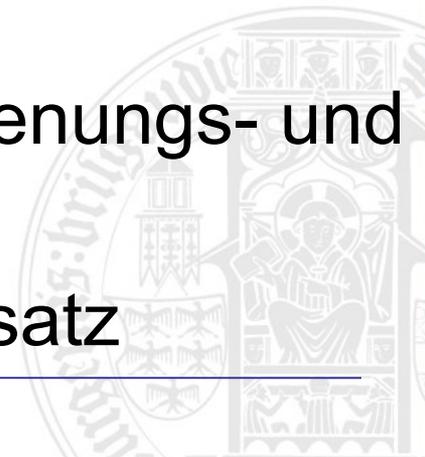
- Performanceverbesserung
 - Multi-Nutzer, persistente Sessions ...
 - Sound wird (noch) nicht übertragen
 - Mauszeiger wird z.T. nicht korrekt abgebildet
 - Schaffung von Schnittstellen, beispielsweise zu OAIS-basierten Archiven
-





Anwendbarkeit und Grenzen I

- Emulation für fast beliebige digitale Objekte geeignet
 - Emulation eher für Interaktion, denn automatisierte Barbeitung großer Objektbestände
 - Zunehmender Abstand zwischen Ursprungs- und Referenzumgebung
 - Auseinanderentwicklung der Bedienungs- und Interaktionsparadigmen
 - Ergänzung von Migration, kein Ersatz
-





Anwendbarkeit und Grenzen II

- Problem DRM
 - Hinterlegung aller potenziell notwendiger Software und Lizenzen bei entsprechenden Gedächtnisorganisationen
 - Entfernung aller Lizenzschutzmaßnahmen
 - DRM-geschützte Objekte in einem LA sind wertlos!
 - Neue Typen webbasierter Applikationen
 - Massive Multiplayer Online Games
-





Fragen!? / Kontakt Information

Vielen Dank für die Aufmerksamkeit!

**Lehrstuhl für Kommunikationssysteme /
Rechenzentrum der Universität
Herrmann-Herder-Str. 10
79104 Freiburg**

Tel. +49 761 203 4698 / 8058

Fax +49 761 203 4640

dsuchod@rz.uni-freiburg.de

rwelte@rz.uni-freiburg.de

www.ks.uni-freiburg.de

