

## Jahresbericht 2009 der Gruppe „Betriebssysteme und Middleware“

<http://www.dcl.hpi.uni-potsdam.de/Jahresbericht2009.pdf>

### 1. Personelle Zusammensetzung

#### Leiter

Prof. Dr. rer. nat. habil. Andreas Polze  
Tel.: (0331) 5509 231  
email: [andreas.polze@hpi.uni-potsdam.de](mailto:andreas.polze@hpi.uni-potsdam.de)

#### Sekretariat

Sabine Wagner  
Tel: (0331) 5509 220  
Fax: (0331) 5509 229  
email: [sabine.wagner@hpi.uni-potsdam.de](mailto:sabine.wagner@hpi.uni-potsdam.de)

#### Wissenschaftliche Mitarbeiter

Dr. rer. nat. Martin von Löwis  
Tel.: (0331) 5509 239  
email: [loewis@hpi.uni-potsdam.de](mailto:loewis@hpi.uni-potsdam.de)

Dr. rer. nat. Wolfgang Schult  
Tel.: (0331) 5509 234  
email: [wolfgang.schult@hpi.uni-potsdam.de](mailto:wolfgang.schult@hpi.uni-potsdam.de)

Dipl.-Inf. Bernhard Rabe  
Tel.: (0331) 5509 236  
email: [bernhard.rabe@hpi.uni-potsdam.de](mailto:bernhard.rabe@hpi.uni-potsdam.de)

Dipl.-Inf. Alexander Schmidt  
Tel.: (0331) 5509 238  
email: [alexander.schmidt@hpi.uni-potsdam.de](mailto:alexander.schmidt@hpi.uni-potsdam.de)

Michael Schöbel, M.Sc.  
Tel.: (0331) 5509 109  
email: [michael.schoebel@hpi.uni-potsdam.de](mailto:michael.schoebel@hpi.uni-potsdam.de)

Johannes Passing, M.Sc.  
email: [johannes.passing@hpi.uni-potsdam.de](mailto:johannes.passing@hpi.uni-potsdam.de)

Frank Feinbube, M.Sc.  
Tel.: (0331) 5509 235  
email: [frank.feinbube@hpi.uni-potsdam.de](mailto:frank.feinbube@hpi.uni-potsdam.de)

Dipl.-Ing. (FH) Uwe Hentschel, M.Sc.  
Tel.: (0331) 5509 237  
email: [uwe.hentschel@hpi.uni-potsdam.de](mailto:uwe.hentschel@hpi.uni-potsdam.de)

Jan-Arne Sobania, M.Sc.  
Tel.: (0331) 5509 237  
email: [jan-arne.sobania@hpi.uni-potsdam.de](mailto:jan-arne.sobania@hpi.uni-potsdam.de)

Robert Wierschke, M.Sc.  
Tel: (0331) 5509 214  
email: [robert.wierschke@hpi.uni-potsdam.de](mailto:robert.wierschke@hpi.uni-potsdam.de)

Dipl.-Inf. Theodor Heinze  
Tel: (0331) 5509 214  
email: [theodor.heinze@hpi.uni-potsdam.de](mailto:theodor.heinze@hpi.uni-potsdam.de)

Dipl.-Inf. Alexander Schacht  
Tel: (0331) 5509 233  
email: [alexander.schacht@hpi.uni-potsdam.de](mailto:alexander.schacht@hpi.uni-potsdam.de)

## Unsere Mission

Moderne Betriebssysteme entwickeln sich in mehrere Richtungen. Einfache Benutzbarkeit, graphische Oberflächen, dynamische Erkennung von Geräten, Unterstützung verteilter Rechnens - dies sind Charakteristika moderner Desktop-Betriebssysteme, wie Windows XP, Mac OS X oder Linux. Noch rasanter entwickelt sich Middleware - eine Softwareschicht oberhalb der Betriebssystemebene. Häufig finden sich dort Dienste und Funktionen, die applikationsübergreifend benötigt werden und (noch) nicht Eingang in die Betriebssysteme gefunden haben. Im Laufe der Zeit werden Middleware-Dienste dann zu Bestandteilen der Betriebssysteme.

Handelsübliche Computer-Systeme (COTS - Commercial off-the-shelf) sind für hohe Performance im lokalen Fall optimiert. Dies führt häufig zu unbefriedigendem Verhalten beim Einsatz in weit-verteilten Szenarien im Internet. Vorhersagbares Ende-zu-Ende Verhalten von verteilten Echtzeitdiensten ist daher eine zentrale Forschungsaufgabe, die sich für Betriebssysteme und Middleware-Plattformen stellt.

**Die Forschungsaktivitäten der Gruppe „Betriebssysteme und Middleware“ konzentrieren sich auf Paradigmen, Entwurfsmuster, und Implementierungsansätze für vorhersagbares Rechnen in Middleware-Umgebungen. Von zentralem Interesse ist dabei die Fragestellung, wie weit der Einsatz von Middleware-Technologie in der Domäne der eingebetteten Systeme vorangetrieben werden kann. Das Ziel der Arbeiten ist die Verbindung von eingebetteten Steuerungssystemen mit Standard-Middleware.**

Eine Reihe von Forschungsprojekten behandelt zudem das Konfigurationsproblem für Komponentensoftware. Insbesondere geht es um die dynamische Ersetzung von Komponenten (online replacement of software components) und die Benutzung von handelsüblichen Betriebssystemen für Steuerungsaufgaben. Für unsere Werkzeuge setzen wir Techniken der aspektorientierten Programmierung (AOP) ein.

## 2. Lehrveranstaltungen

### 2.1. Vorlesungen

- Betriebssysteme I (WS 09/10)
- Betriebssystem II (SS 09)
- Programmiertechnik I (WS 09/10)
- Programmiertechnik II (SS 09)
- Compilerbau mit Phoenix (WS 08/09)
- Datenorientiertes XML (SS 09)
- Middleware und Verteilte Systeme (WS 09/10)

### 2.2. Übungen

- Betriebssysteme I (WS 09/10)
- Betriebssysteme II (SS 09)
- Programmiertechnik I (WS 09/10)
- Programmiertechnik II (SS 09)

### 2.3. Seminare/Praktika

- Forschungsseminar der Gruppe „Betriebssysteme und Middleware“
- Bachelorvorbereitungsseminar (WS 08/09)
- Betriebssystemadministration (SS 09)
- Entwicklungsprozesse in Open-Source-Projekten (WS 09/10)
- Studienbegleitendes Seminar/Seminarorganisation (WS 08/09)
- Quellcode-Analyse Seminar (SS 09)

### 2.4. Teleteaching

- Aufzeichnung der Veranstaltungen Programmiertechnik I und II, „Datenorientiertes XML“ und „Middleware und Verteilte Systeme“

### 3. Betreuung von Studierenden und Dissertationen

#### 3.1. Betreuung von Bachelorprojekten

- Entwicklung eines Tools zur modellbasierten Spezifikation auf Basis von Eclipse und GEF/EMF, Bachelorprojekt mit der Firma Biotronik, Berlin (WS 09/10)
- Softwareentwicklung bei der Deutschen Post IT-Services Berlin (SS 09)

#### 3.2. Betreuung von Masterarbeiten

- Jan-Arne Sobania – A Tracing Infrastructure for the Windows Research Kernel, Potsdam 2009
- Frank Feinbube. Performance Evaluation in Application Servers
- Daniel Richter. Analyse des Ressourcenverbrauchs in konsolidierten Umgebungen am Beispiel betriebswirtschaftlicher Standardsoftwaresysteme
- Robert Wierschke. Analyse des Ressourcenverbrauchs in virtualisierten Systemen - Am Beispiel der E/A-Last betriebswirtschaftlicher Standardanwendungen

#### 3.3. Betreuung von Dissertationen (intern, extern)

- Bernhard Rabe – Programmoptimierung für eingebettete Systeme
- Wolfgang Schult – Architektur komponentenbasierter Systeme mit LOOM: Aspekte, Muster, Werkzeuge (eingereicht am 19.12.2008, verteidigt am 25.05.2009)
- Michael Schöbel – Verarbeitung von Ereignisströmen im Betriebssystemkern, im Forschungskolleg „Service-Oriented Systems Engineering“
- Alexander Schmidt – Ressourcenverwaltung in Betriebssystemen, im Forschungskolleg „Service-Oriented Systems Engineering“
- Frank Feinbube – Programmiermodelle für Multicore Computer
- Uwe Hentschel – Services for Real-Time Computing
- Jan-Arne Sobania – Statische Analyse von Programmen für eingebettete Systeme
- Alexander Schacht – „Echtzeitsysteme in der Telemedizin“, im Verbundprojekt „SaPiMa/PhysioGate“ / FONTANE
- Robert Wierschke – „Kommunikationsmiddleware für telemedizinische Systeme“, im Verbundprojekt „SaPiMa/PhysioGate“ / FONTANE
- Theodor Heinze – „Regelbasierte Entscheidungsunterstützung in der Telemedizin“, im Verbundprojekt „SaPiMa/PhysioGate“ / FONTANE

## 4. Bearbeitete Forschungsthemen

### 4.1. Lego.NET

**Ansprechpartner: Dr. Martin v. Löwis**

Ziel des Projekts Lego.NET ist es zu untersuchen, inwieweit Softwareentwicklung für Lego-Mindstorm-Roboter und ähnliche Geräte auf Basis von Microsofts .NET möglich ist.

Dazu wurde im Rahmen des Projekts die GNU Compiler Collection (gcc) erweitert und ein neues Front-End realisiert, mit dem .NET Common Intermediate Language (CIL) in den Maschinencode des Prozessors im Mindstorm-Roboter (Renesas H8/300) übersetzt wird. Die so übersetzten Programme können auf Basis des Betriebssystems brickOS ausgeführt werden.

Gegenwärtig unterstützt die Version 1.4 von Lego.NET zentrale Sprachkonzepte von C# und CIL, wie Kontrollflußkonstrukte, Klassen, Vererbung und Polymorphie, Aufzählungstypen und Delegates. Andere Aspekte wie Speicherverwaltung, Ausnahmebehandlung und Fragen der Codeoptimierung sind Thema der Vorlesung „Compilerbau für die Common Language Run-Time“ sowie von Master-Arbeiten.

### 4.2. Micro.NET

**Ansprechpartner: Dipl.-Inf. Bernhard Rabe**

In der Softwareentwicklung für eingebettete Systeme haben lange Zeit hardwarenahe Programmiersprachen dominiert. Virtuelle Ausführungsumgebungen wie Java VM und die Common Language Infrastructure (CLI) haben die Softwareentwicklung durch automatische Speicherverwaltung, Typsicherheit, Ausnahmebehandlung und Plattformunabhängigkeit vereinfacht. Im Rahmen des Micro.NET Projektes werden Techniken untersucht und entwickelt um .NET Technologie an die Anforderungen eingebetteter Systeme anzupassen. Betrachtet werden insbesondere Systeme deren spezielle Eigenschaften und Anforderungen von bestehende CLI Implementierungen (.NET, .NET CF, .NET MF, Mono) nicht berücksichtigt werden. Mit dem entstandenen Bytecode-Interpreter für die Lego Mindstorm Plattform konnte die Ausführbarkeit von .NET auf ressourcenbeschränkten eingebetteten Systemen gezeigt werden. Um Speicheranforderungen von .NET Programmen zu minimieren wurde das Konzept der self-contained Assemblies entwickelt. Damit ist es möglich die lose Kopplung zwischen .NET Programmen und Bibliotheken zugunsten eines geringeren Speicherplatzbedarfs aufzuheben, indem alle benötigten Bibliotheksfunktionen in das Programm integriert werden. Basierend auf dem ECMA 335 Standards werden Methoden zur Spezifikation und Entwicklung einer minimalen Laufzeitumgebung für Self-contained Assemblies untersucht.

### 4.3. Das LOOM.NET Projekt

**Ansprechpartner: Dr. Wolfgang Schult**

Das Konzept der Aspekt-Orientierten Programmierung ermöglicht einen interessanten Ansatz zur Modellierung und Implementierung nichtfunktionaler Komponenteneigenschaften. Diese nichtfunktionalen Eigenschaften werden als sogenannte Aspekte mit Hilfe eines Aspektwebers in den Komponentencode eingewoben. Am Fachgebiet Betriebssysteme und Middleware gibt es seit einigen Jahren das Loom.Net Projekt, welches sich mit der

Entwicklung solcher Aspektweber auf der Microsoft.NET Plattform beschäftigt. Derzeit gibt zwei Aspektweber die im Rahmen dieses Projektes ständig weiterentwickelt werden.

Rapier-Loom.NET ist ein dynamischer Aspektweber der als Klassenbibliothek ohne spezielle Kompiler oder Änderung der Ausführungsumgebung aspektorientiertes Programmieren auf der .NET-Plattform ermöglicht. Dabei stützt sich die Aspektbeschreibungssprache vollständig auf den ECMA-Standard.

Gripper-Loom.NET ist der neueste Aspektweber des LOOM.NET-Projektes. Basierend auf Microsoft's Phoenix-Kompiler-Framework ist Gripper-Loom.NET ein statischer Aspektweber, der als zusätzlicher Build-Step die Aspekte verwebt. Die Aspektbeschreibung ist identisch zu Rapier-Loom.NET, sodass es möglich ist, dort definierte Aspekte auch mit Gripper-Loom.NET zu verweben.

Zum Einsatz kommen die Aspektweber in weiteren Projekten des Lehrstuhls wie dem Distributed Control Lab oder Adapt.NET. Aber auch außerhalb der Forschungstätigkeiten werden die LOOM.NET Aspektweber im kommerziellen Projekten eingesetzt. Auf der PRIO-2007, einer Konferenz für Architekten und Entwickler auf der Microsoft .NET-Plattform aus dem industriellen Umfeld wurde die Umsetzung des Design-by-Contract Konzeptes mit LOOM.NET erfolgreich demonstriert.

Die aktuellen Versionen der Aspektweber können auf der offiziellen Loom.Net Homepage ([www.rapier-loom.net](http://www.rapier-loom.net)) heruntergeladen werden.

#### 4.4. Windows Research Kernel / Windows Monitoring Kernel

**Ansprechpartner: Michael Schöbel, M.Sc., Dipl.-Inf. Alexander Schmidt**

Unter dem Namen „Windows Operating System Internals Curriculum Resource Kit“ (CRK) entstand 2005 eine Lehrmaterialsammlung zum Thema Betriebssystemarchitektur. Das CRK ist eng mit dem „Windows Research Kernel“ (WRK) verknüpft, einer Plattform, die Modifikationen am Windows-Kern ermöglicht.

Forschungsfragen, wie neue Scheduling-Algorithmen und Bandbreiten-Reservierungen im Ein-/Ausgabesystem lassen sich so untersuchen. Dazu entstanden verschiedene Testanwendungen und Kernmodifikationen. Außerdem wurde das Open-Source Werkzeug Doxygen zur Aufbereitung des Quellcodes verwendet.

Auf Basis des Windows Research Kernels wurde ein Framework entwickelt, das eine Black-Box-Analyse von Anwendungen erlaubt. Dazu wurde der Windows-Kern um ein effizientes Event-Log erweitert und geeignet instrumentiert. Außerdem wurden neue Schnittstellen eingeführt, mit deren Hilfe auch Ereignisse in verschiedenen Bibliotheken aufgezeichnet werden können.

#### 4.5. KStruct – Live Kernel Data Inspection

**Ansprechpartner: Dipl.-Inf. Alexander Schmidt**

**Beteiligt: Jan-Arne Sobania, M.Sc., Lysann Kessler, Frank Schlegel**

Im Mittelpunkt des Projekts steht der Versuch, den Prozess für das Verstehen eines komplexen Softwaresystems zu vereinfachen. Ein Entwickler verfügt typischerweise über den Quellcode sowie die Dokumentation des Softwaresystems, in das er sich einarbeiten möchte. Dies reicht jedoch nicht aus, um das System gänzlich zu verstehen. Vielmehr wäre es wünschenswert, den Zustand des Systems, d.h. Exemplare verwendeter Datenstrukturen, zur Laufzeit zu beobachten, um daraus weitere Schlüsse zu ziehen. Eine solche Beobachtung

kann beispielsweise durch die Verwendung von Debuggern oder Instrumentierung erfolgen. Als Beispiel für ein solches komplexes Softwaresystem betrachten wir den Windows Research Kernel.

Die Idee hinter KStruct ist es, Datenstrukturen, die beobachtet werden sollen, semantisch so zu beschreiben, dass ein Treiber generiert werden kann, der ein Exemplar der Datenstruktur lokalisieren und sicher referenzieren kann. Die Beschreibung der Datenstrukturen erfolgt mit einer C-ähnlichen Syntax.

## 4.6. Eingebettete Systeme im Distributed Control Lab

**Ansprechpartner: Dipl.-Inf. Bernhard Rabe**

Im Distributed Control Lab wird die Verbindung von Middleware-basierten Komponenten und eingebetteten Systemen untersucht. Ziel ist es vorhersagbares Systemverhalten in instabilen Umgebungen zu erreichen. Zur Untersuchung von verschiedenen Ansätzen sind Fallstudien in Form von Experimenten umgesetzt, die sich über das Internet steuern lassen. Im Vordergrund stehen dabei die nichtfunktionalen Eigenschaften, wie z.B.: Echtzeitverhalten, Fehlertoleranz und Sicherheit.

Der Schutz der Experimente (Hardware) vor potentiell fehlerhaften Kontrollalgorithmen aus dem Internet wird abhängig von den Anforderungen der Experimente mittels Soft- bzw. Hardwaremechanismen realisiert.

Als wichtiger Pfeiler für praktische Erfahrungen floss die verteilte Laborinfrastruktur in die Lehrveranstaltung „Betriebssysteme für Embedded Computing“ ein. Dabei wurden in verschiedenen Praktikumsveranstaltungen die Steuerung eines Intelligenten Hauses mit Beckhoff Soft-SPS (speicherprogrammierbare Steuerungen) realisiert, die Eignung der .NET Plattform für eingebettete Geräte (.NET Compact Framework) wurde an Hand des „Hau-den-Lukas“-Experiments evaluiert. Auch die Steuerung einer komplexen Fischertechnik Fertigungsstraße mit Hilfe von industriellen Steuerungssystemen wurde im Rahmen eines Bachelorprojekts in das Distributed Control Lab integriert und verschiedene Sicherheitsstrategien implementiert. Zusätzlich kam die Laborinfrastruktur als Lernumgebung für eine Veranstaltung zum Thema „Programmierung eingebetteter Systeme“ an der Blekinge Tekniska Högskola in Ronneby, Schweden zum Einsatz. Studenten der Veranstaltung führten physikalisch entfernte Experimente am Hasso-Plattner-Institut als Praktikumsaufgaben durch, um das in der Vorlesung erworbene Wissen zu vertiefen.

Im EU-Projekt Vet-Trend wird untersucht wie, die Laborinfrastruktur in weitere Veranstaltungen auf europäischer Ebene integriert werden kann und auch wie existierende Infrastrukturen der Projektpartner am HPI nutzbar gemacht werden können. Zusätzlich umfasst das Projekt Standardisierungsbestrebungen auf dem Gebiet der verteilten und virtuellen Labore

## 4.7. Real-Time.Net

**Ansprechpartner: Dr. Martin v. Löwis**

**Beteiligt: Sven Widmer**

Virtuelle Laufzeitumgebung wie Java und Microsoft .NET haben die Entwicklung von Software in den letzten Jahren verbessert. Automatische Speicherverwaltung, erhöhte Typsicherheit und umfangreiche Klassenbibliotheken reduzierten die Entwicklungszeit von Softwaresystemen. Diese Vorteile sind auch für die Entwicklung von eingebetteten Systemen wünschenswert. Um eine deterministische Ausführung von Programmcode und die

Implementierung von Hardware-nahen Treibern zu ermöglichen, müssen die vorhandenen Ansätze erweitert werden. Im Projekt wird die Integration solcher Konstrukte in die .NET Plattform untersucht. Dazu zählen eine deterministische Speicherverwaltung, uneingeschränkter Hardwarezugriff, Unterbrechungsbehandlung, zeitbasierende Programmausführung sowie deterministische Ausführung von Programmabschnitten selbst.

Ausgehend vom Lego.Net Compiler-Ansatz wurde im Projekt ein Laufzeitsystem für das Echtzeitbetriebssystem Windows CE.NET implementiert und an Experimenten im Distributed Control Lab evaluiert.

#### 4.8. Dynamic Software Update Platform (DSUP)

**Ansprechpartner: Dipl.-Inf. Wolfgang Schult**

Um die Ausfallzeit von Software beim Einspielen neuer Softwareversionen zu minimieren ist es von Vorteil Aktualisierungen während der Laufzeit einzuspielen. Die Reaktion auf neue Sicherheitslücken, Anpassungen an wechselnde Umgebungseigenschaften sowie neue Kundenwünsche können auf diese Weise schneller integriert werden. Im Projekt wurde die Filialsoftware der Deutschen Post untersucht und ein existierender Updatemechanismus verbessert. Hierbei wurde mit Hilfe von Techniken der Aspektorientierten Programmierung ein Algorithmus entwickelt, welcher das Einspielen neuer Softwareversionen während der Laufzeit einer Anwendung realisiert. Neben der Überführung der Software in einen aktualisierbaren Zustand wurde ein graphentheoretischer Algorithmus zum Zustandstransfer zwischen alter und neuer Version implementiert. Die implementierten Algorithmen werden weiterhin in der Adapt.Net-Infrastruktur für die Konfiguration aktiver Komponenten verwendet.

#### 4.9. IPv6 Testbed

**Ansprechpartner: Dr. Martin von Löwis**

Nach verschiedenen Schätzungen von APNIC, ARIN, LACNIC und RIPE aus dem Jahr 2007 werden die regionalen Registrierungen ihren Vorrat an IPv4-Adressen im Laufe des Jahres 2010 erschöpfen. Gleichzeitig gibt es aber keine ausreichenden Bestrebungen, Internetnutzern den Zugang zum IPv6-Wirknetz bereitzustellen, und kein ausreichendes Interesse dieser Nutzer, von bestehenden IPv6-Angeboten Gebrauch zu machen.

Ziel des IPv6-Testbeds am HPI ist, Studenten, Lehrkräften und Technikern eine realistische Umgebung bereitzustellen, um die Auswirkungen von IPv6 auf Netzbetrieb und Softwareentwicklung zu studieren. Dieses Netz ist an das globale IPv6-Netz angeschlossen, so dass praxisnahe Arbeiten mit IPv6 und den dazugehörigen Technologien und Protokollen möglich ist.

#### 4.10. PXR – Ein HTML-basierter Dokumentations-Compiler

**Ansprechpartner: Dipl.-Inf. Alexander Schmidt**

**Beteiligte: Lysann Kessler, Frank Schlegel**

Für das Verstehen von Quellcode in beliebig komplexen Softwaresystemen ist es hilfreich, den Quellcode strukturiert aufzubereiten und durch zusätzliche Informationen anzureichern. Wird der Quellcode im Rahmen eines Teams analysiert, bspw. im Rahmen einer Vorlesung, stellt sich des Weiteren das Problem der Verteilung der Quellen.

Dem PXR-Projekt liegt die Idee zu Grunde, den Quellcode des Windows Research Kernels in HTML umzuwandeln auf einem Webserver HPI-intern verfügbar zu machen. Ein ähnlicher Ansatz ist bereits in Tools wie Doxygen oder LXR zu finden. Der Vorteil der Verwendung von HTML liegt darin, dass ein Nutzer sich natürlich durch die Quellen navigieren kann, in dem er Querverweisen folgt. Bei den genannten Tools ist jedoch lediglich ein (C-) Parser im Einsatz, was mit unter zu ungültigen Verweisen oder Ungenauigkeiten führt. PXR erweitert diesen Ansatz daher, indem zusätzlich Informationen aus dem Compiler extrahiert und in den HTML-Code einbettet. So lässt sich beispielsweise genau herausfinden, an welcher Stelle im Code auf eine bestimmte Variable zugegriffen wird oder an welcher Stelle einer Funktion Schreib- bzw. Lesezugriffe auf eine Variable erfolgen.

#### 4.11. Serverbetriebssystem-Labor

**Ansprechpartner: Dr. Martin von Löwis, Dipl.-Inf. Bernhard Rabe**

**Beteiligte: Martin Kreichgauer**

An Serverbetriebssysteme werden höhere Anforderung an Fehlertoleranz, Verfügbarkeit, Sicherheit und Skalierbarkeit gestellt im Vergleich zu Desktop-Betriebssysteme. In diesem Zusammenhang untersuchen wir Techniken für effiziente Betriebsmittelverwaltung, Sicherheitskonzepte und Hochverfügbarkeitsansätze. Virtualisierung ist ein Ansatz in Serverbetriebssystemen um Betriebsmittel besser auszunutzen, Isolation von Betriebssysteminstanzen zu realisieren und Betriebsmittelbenutzung zu regeln. Wir betreiben Virtualisierungslösungen von Microsoft (Hyper-V), Hewlett-Packard (HP-VM), VMWare (ESX), Sun (Solaris Zones) und Linux 2.6 (KVM) auf x64, IA64 und Sparc-Architektur.

Für Fehlertoleranz- und Hochverfügbarkeitsuntersuchungen betreiben wir mehrere OpenVMS-Cluster auf verschiedenen Architekturen (IA64, AXP, VAX). Serverbetriebssystemkonzepte sind Bestandteil in Forschung und Lehre und werden in den Vorlesungen Betriebssysteme I+II und Serverbetriebssysteme vermittelt, sowie für tiefgehenden Untersuchungen im Rahmen von Masterarbeiten behandelt.

#### 4.12. Grenzen von Betriebssystemressourcen

**Ansprechpartner: Frank Feinbube, M.Sc.**

**Beteiligte: Fabian Bornhofen, Thomas Büniger**

Das Ressourcenmanagement von Betriebssystemen ist traditionell auf die Prozessabstraktion als Basisgröße für die Granularität ausgerichtet. Die simple Lösung einen Prozess (oder eine virtuelle Maschine) einem Prozessorkern zuzuweisen muss mit dem Aufkommen von Mehrkernarchitekturen und der Unterstützung von Hyperthreading neu beurteilt werden. Wir erforschen die Begrenzung des Threadmanagements auf diversen Betriebssystemen (Windows Server 2003, Windows Server 2008 R2, HP-UX, Solaris, Ubuntu). Neben der Anzahl von Threads, die ein Betriebssystem unterstützt, werden auch die Fragen nach dem Einfluss auf den verwendeten Hauptspeicher, die Zeit zur Threaderzeugung und der Einfluss auf den Schedulingoverhead untersucht.

Bei der Diskussion der Architektur komplexer Softwaresysteme in Hinblick auf die Verwendung von Threads oder Prozessen als grundlegende Mechanismen der Strukturierung müssen auch weitere Qualitätsmerkmale wie Zugriffsschutz, Isolation und Fehlereingrenzung berücksichtigt werden. Weiterhin untersuchen wir inwiefern strukturierte Ausnahmebehandlung (SEH) unter Windows als ein Mechanismus verwendet werden kann, um in einem Thread-basierten System einen Zugriffsschutz und eine Isolation zu erreichen, die mit denen von traditionellen Prozess-basierten System vergleichbar ist.

#### 4.13. Fontane – Gesundheitsregionen der Zukunft / Nordbrandenburg

**Ansprechpartner: Dr. Martin von Löwis, Uwe Hentschel, M.Sc.**

Das Fontane Projekt konzentriert sich auf die Fernüberwachung und die Nachsorge für Schlaganfall-Patienten und Patienten mit Herzkrankheiten im ländlichen Raum von Nordbrandenburg. Das Projekt wird von einem Konsortium von 20 Partnern durchgeführt, darunter die Charité, die größte Universitätsklinik in Deutschland, Hersteller von medizinischen Geräten, Ärzte und Herz-Spezialisten. Für Fontane entwickelt und implementiert die Fachgruppe „Betriebssysteme und Middleware“ des HPI eine selbstadaptive, priorisierende Middleware, die auf zuvor entwickelten Konzepten von objektbasierten Echtzeit-Systemen aufbaut.

Das Fontane-Projekt baut auf den Erfahrungen des Vorgänger-Projektes „Partnership for the Heart“ auf, das das Problem der Überwachung von 300 Patienten im Großraum Berlin gelöst hat. Im Gegensatz zu dem Vorgänger-Projekt und mit mehr als 1000 teilnehmenden Patienten, wird Fontane mit Fragen der Skalierbarkeit konfrontiert werden, die neue Ansätze für die Priorisierung der Übertragung und Darstellung von Daten erfordern. Ein weiteres Novum der von uns vorgeschlagene Architektur wird ihre allgemeine Gültigkeit sein: Dank ihrer Selbstadaptivität und ihrer regelbasierten inneren Abläufe wird sie nicht nur zur Überwachung von Herz-Patienten, sondern auch zur Versorgung von Patienten mit anderen Krankheiten geeignet sein.

## 5. Drittmittelprojekte

- Förderung für Teilprojekt SaPiMa / PhysioGate im Rahmen des Fontane-Projektes durch die Zukunftsagentur Brandenburg
- Fördersumme: 1,2 Mio EUR, 5 Doktoranden-/Mitarbeiterstellen

## 6. Forschungsk Kooperation (außerhalb des HPI)

- Microsoft Corp., Redmond, Windows Academic Team
- Getemed Medizintechnik AG, Teltow

## 7. Publikationen

### 7.1. Begutachtete Konferenzartikel

- Alexander Schmidt.  
Automatic Extraction of Locking Protocols.  
In *Proceedings of the 4<sup>th</sup> Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering*. Potsdam, Germany, October 2009.  
Technischer Bericht des Hasso-Plattner-Instituts für Softwaresystemtechnik an der Universität Potsdam, Heft 31 (2010), ISBN 978-3-86956-036-6
- Johannes Passing, Alexander Schmidt, Martin von Löwis, and Andreas Polze.  
Ntrace: Function Boundary Tracing for Windows on IA-32.  
In *Proceedings of the 16th Working Conference on Reverse Engineering*. October 13-16, 2009, Lille, France.
- Alexander Schmidt, Martin von Löwis, and Andreas Polze.  
KStruct: Preserving Consistency Through C Annotations.  
In *Proceedings of the 5th Workshop on Programming Languages and Operating Systems*. October 11, 2009, Big Sky, Montana, USA.
- Michael Schöbel.  
A Runtime Environment for Online Processing of Operating System Kernel Events.  
In *Proceedings of the 4<sup>th</sup> Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering*. Potsdam, Germany, October 2009.  
Technischer Bericht des Hasso-Plattner-Instituts für Softwaresystemtechnik an der Universität Potsdam, Heft 31 (2010), ISBN 978-3-86956-036-6
- Frank Feinbube  
On Programming Models for Multi-Core Computers.  
In *Proceedings of the 4<sup>th</sup> Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering*. Potsdam, Germany, October 2009.  
Technischer Bericht des Hasso-Plattner-Instituts für Softwaresystemtechnik an der Universität Potsdam, Heft 31 (2010), ISBN 978-3-86956-036-6
- Michael Schöbel, Andreas Polze.  
A Runtime Environment for Online Processing of Operating System Kernel Events.

In *Proceeding of the 7<sup>th</sup> International Workshop on Dynamic Analysis*. 2009, Chicago, IL, USA

- Dawit Mengistu, Martin v. Löwis.  
An Algorithm for Optimized Distributed Simulations.  
In *Proceedings of the IASTED conference on Modelling, Simulation and Identification (MSI 2009)*, Peking, 2009
- Paul Bouché, Martin von Löwis, Peter Tröger.  
Aggregated Accounting of Memory Usage in Java.  
In *Proceedings of the 4<sup>th</sup> international conference on Software and Data Technologies (ICSOF 2009)*, S. 177-185

## 7.2. Bücher

- Wolfgang Schult. Dissertationsschrift  
Architektur komponenten-basierter Systeme mit LOOM: Aspekte, Werkzeuge, Muster  
Cuvillier Verlag, Juli 2009. ISBN 9783869550312

## 7.3. Technische Berichte / Standardisierungsdokumente

- "Proceedings of the 4th Ph.D. Retreat of the HPI Research School on Service-oriented Systems Engineering"  
Herausg. Meinel, C. , Plattner, H. , Döllner, J. , Weske, M. , Polze, A. , Hirschfeld, R. , Naumann, F. , Giese, H.  
Technische Berichte Nr. 31 des Hasso-Plattner-Instituts für Softwaresystemtechnik an der Universität Potsdam ISBN 978-3-86956-036-6

## 8. Vorträge

### 8.1. Vorträge auf Tagungen

- Johannes Passing. Ntrace: Function Boundary Tracing for Windows on IA-32. 16th Working Conference on Reverse Engineering. October 13-16, 2009, Lille, France
- Alexander Schmidt. KStruct: Preserving Consistency Through C Annotations. 5th Workshop on Programming Languages and Operating Systems. October 11, 2009, Big Sky, Montana, USA
- Alexander Schmidt. Automatic Extraction of Locking Protocols. 4<sup>th</sup> Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering. October 16, 2009, Templin (Groß Dölln), Germany.
- Michael Schöbel. A Runtime Environment for Online Processing of Operating System Kernel Events. 4<sup>th</sup> Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering. October 16, 2009, Templin (Groß Dölln), Germany.
- Michael Schöbel. A Runtime Environment for Online Processing of Operating System Kernel Events. 7<sup>th</sup> International Workshop on Dynamic Analysis, July 20, 2009, Chicago, IL, USA
- Michael Schöbel. Online Processing of Operating System Kernel Events. Workshop – HPI Research School @ SAP, December 9, 2009, Walldorf, Germany

- Uwe Hentschel. Services for Real-Time Computing. 4<sup>th</sup> Ph.D. Retreat of the HPI Research School on Service-Oriented Systems Engineering. October 16, 2009, Templin (Groß Dölln), Germany.
- Martin v. Löwis. IPv6 im NTP-Pool. 1. IPv6-Kongress, 28. Mai 2009, Frankfurt

## 8.2. Vortragseinladungen außerhalb des HPI

- Andreas Polze, Alexander Schmidt, Arkady Retik, and Dave Probert. Windows Kernel Source in the Classroom: Worldwide Use and New Resource Kit. 40th ACM Technical Symposium on Computer Science Education. March 4-7, 2009, Chattanooga, Tennessee, USA.
- Bernhard Rabe. OpenVMS Virtualisierung: VMS Frühjahrstreffen, Bad Homburg, 19. März 2009
- Martin v. Löwis. Portierung von ZODB auf Python 3.1. Keynote, 10. DZUG-Tagung, 9. September 2009, München

## 9. Herausgeberschaft

### 9.1. Websites

- Operating Systems and Middleware Homepage ([www.dcl.hpi.uni-potsdam.de](http://www.dcl.hpi.uni-potsdam.de))
- Adaptive Services Grid ([www.asg-platform.org](http://www.asg-platform.org))
- The Python Programming Language ([www.python.org](http://www.python.org))  
(Mitherausgeber Martin v. Löwis)
- Deutscher IPv6-Rat ([www.ipv6council.de](http://www.ipv6council.de))  
(Mitherausgeber Martin v. Löwis)
- The LOOM.NET Project Page (<http://www.rapier-loom.net>)  
(Herausgeber Wolfgang Schult)
- Windows Research Kernel @ HPI  
(<http://www.dcl.hpi.uni-potsdam.de/research/WRK/>)  
Michael Schöbel, Alexander Schmidt
- Distributed Control Lab  
(<http://tb4.asg-platform.org:8000>)  
Bernhard Rabe, Andreas Rasche

## 10. Mitgliedschaften, Programmkomitees, Gutachtertätigkeiten

### 10.1. Mitgliedschaften

- Prof. Dr. Andreas Polze
  - Mitglied "German Software Architects Council", Microsoft, München.
  - Mitglied IEEE, GI, DECUS (HP User Society)

- Dr. Martin v. Löwis
  - Schatzmeister des SDL Forum e.V.
  - Mitglied im Board of Directors, Python Software Foundation
  - Mitglied im Infrastructure Committee, Python Software Foundation
  - Mitglied im Deutschen IPv6-Rat
- Dr. Wolfgang Schult
  - Mitglied "German Software Architects Council", Microsoft, München.
- Uwe Hentschel, M.Sc.
  - Mitglied VDE Verband der Elektrotechnik Elektronik Informationstechnik e.V.

## 10.2. Mitarbeit in Programmkomitees

- Prof. Dr. Andreas Polze
  - ISORC'09 – Intl. Symposium on Object-ORiented Realtim Systems
  - SAKS – Selbstorganisierende, Adaptive, Kontextsensitive verteilte Systeme
  - ICSOFT – Intl. Conference on Software and Data Technologies
  - ITSIM – International Symposium on Information Technology
- Dr. Martin v. Löwis
  - SDL '09 (14<sup>th</sup> System Design Languages Forum)
  - DLS 2009 (Dynamic Languages Symposium)
- Dipl.-Inf. Bernhard Rabe
  - IEEE Transactions on Learning Technologies 2009

## 10.3. Gutachtertätigkeiten

- Prof. Dr. Andreas Polze:
  - The Computer Journal, Oxford University Press
  - Journal on Systems and Software, Elsevier
  - Journal on Systems Architecture, Elsevier
  - Real-Time Systems Journal, Springer
- Gutachter Microsoft Research Ph.D. programme
- Sprecher HPI Forschungskolleg „Service-Oriented Systems Engineering“

## 10.4. Gastaufenthalte

- Alexander Schmidt – Windows Academic Program, Microsoft Corp. Redmond, WA, August-Oktober 2009

## 11. Workshops

### 4<sup>th</sup> Symposium on Future Trends in Service-Oriented Computing

**18.-19. Juni 2009**

The symposium on "Future Trends in Service-oriented Computing" (FutureSOC) 2009 is held for the fourth time at the Hasso Plattner Institute. FutureSOC highlights the recent work of some Research School members and outlines new trends in the area of Service-oriented Computing.

As the HPI Research School is an interdisciplinary undertaking of the HPI research groups, FutureSOC covers a wide range of topics concerning SOC, like service description, discovery and composition, service deployment, platform configuration and capacity planning, monitoring, service middleware, service-oriented architectures, service management, information as a service, software as a service, service development and maintenance, novel business models for SOAs, economical implications of Web Services and SOAs, service science, mobile and peer-to-peer services, data services, quality of service, exception handling, or service reliability and security.

Excellent speakers, leaders in their respective field of research, are invited to talk about their latest projects and resulting outcomes.