Gesellschaft für Informatik (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in cooperation with GI and to publish the annual GI Award dissertation.

Broken down into the fields of
• Seminar
• Proceedings
• Dissertations
• Thematics
current topics are dealt with from the fields of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure the high level of the contributions.

The volumes are published in German or English.

Information: http://www.gi-ev.de/service/publikationen/lni/

ISSN 1617-5468
ISBN 978-3-88579-243-7

The 2. DFN-Forum Communication Technologies 2009 is taking place in Munich, Germany, from Mai 27th to Mai 28th.

This volume contains 13 papers selected for presentation at the conference.

To assure scientific quality, the selection was based on a strict and anonymous reviewing process.

Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.): 2. DFN-Forum 2009

GI-Edition

Lecture Notes in Informatics

Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)

2. DFN-Forum Kommunikationstechnologien

Beiträge der Fachtagung

27. Mai bis 28. Mai 2009
München

Proceedings

Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)

2. DFN-Forum Kommunikationstechnologien

Verteilte Systeme im Wissenschaftsbereich

27.05. - 28.05.2009
in München

Gesellschaft für Informatik e.V. (GI)

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume P-149

ISBN 978-3-88579-243-7
ISSN 1617-5468

Volume Editors
Prof. Dr. Paul Müller
Technische Universität Kaiserslautern
Postfach 3049, 67653 Kaiserslautern
Email: pmueller@informatik.uni-kl.de
Prof. Dr. Bernhard Neumair
Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen
Am Fassberg, 37077 Göttingen
Email: Bernhard.Neumair@gwdg.de
Prof. Dr. Gabi Dreo Rodosek
Universität der Bundeswehr München
Werner-Heisenberg-Weg 39, 85577 Neubiberg
Email: Gabi.Dreo@unibw.de

printed by Köllen Druck+Verlag GmbH, Bonn

Vorwort

Der DFN-Verein ist seit seiner Gründung dafür bekannt, neueste Netztechnologien und innovative netznahe Systeme einzusetzen und damit die Leistungen für seine Mitglieder laufend zu erneuern und zu optimieren. Beispiele dafür sind die aktuelle Plattform des Wissenschaftsnetzes X-WiN und Dienstleistungen für Forschung und Lehre wie die DFN-PKI und DFN-AAI. Um diese Technologien einerseits selbst mit zu gestalten und andererseits frühzeitig die Forschungsergebnisse anderer Wissenschaftler kennenzulernen, veranstaltet der DFN-Verein seit vielen Jahren wissenschaftliche Tagungen zu Netztechnologien. Mit den Zentren für Kommunikation und Informationsverarbeitung in Forschung und Lehre (ZKI) e.V. gibt es in diesem Bereich eine langjährige und fruchtbare Zusammenarbeit.

Das 2. DFN-Forum Kommunikationstechnologien „Verteilte Systeme im Wissenschaftsbereich“ steht in dieser Tradition. Nach dem sehr erfolgreichen 1. DFN-Forum im Mai 2008 in Kaiserlautern wird die diesjährige Tagung vom DFN-Verein und dem Leibniz-Rechenzentrum gemeinsam mit dem ZKI e.V. und der Universität der Bundeswehr München am 27. und 28. Mai 2009 in München veranstaltet und soll eine Plattform zur Darstellung und Diskussion neuer Forschungs- und Entwicklungsergebnisse aus dem Bereich TK/IT darstellen. Das Forum dient dem Erfahrungsaustausch zwischen Wissenschaftlern und Praktikern aus Hochschulen, Großforschungseinrichtungen und Industrie.

Aus den eingereichten Beiträgen konnte ein hochwertiges und aktuelles Programm zusammengestellt werden, das neben künftigen Netztechnologien unter anderem auf Grid-Anwendungen, Identity Management und Rechenzentrumstechnologien in Hochschulen eingeht und auch Beiträge aus der Industrie enthält. Ergänzt wird es durch eine Podiumsdiskussion zur künftigen Architektur des Internet (Future Internet) und durch eingeladene Beiträge zu IT-Technologien in Fahrzeugen, zu Cloud Computing und zu IT-Sicherheitsaspekten. Um den Rahmen der Veranstaltung nicht zu sprengen, konnten leider nur etwa die Hälfte der eingereichten Beiträge angenommen werden. Dies zeigt, dass die Veröffentlichung der Beiträge sowohl im Rahmen der GI-Edition Lecture Notes in Informatics als auch Open Access für die Wissenschaftlerinnen und Wissenschaftler attraktiv ist.

Wir möchten uns bei den Autoren für alle eingereichten Beiträge und beim Programmkomitee für die Auswahl der Beiträge und die Zusammenstellung des Programms bedanken. Allen Teilnehmer wünschen wir für die Veranstaltung interessante Vorträge und fruchtbare Diskussionen.

München, April 2009

Paul Müller
Bernhard Neumair
Gabi Dreo Rodosek

Programmkomitee

Alexander Clemm, Cisco

Gabi Dreo Rodosek (Co-Chair), Universität der Bundeswehr München

Thomas Eickermann, Forschungszentrum Jülich

Markus Fidler, Technische Universität Darmstadt

Alfred Geiger, T-Systems SfR

Wolfgang Gentzsch, DEISA

Hannes Hartenstein, Universität Karlsruhe

Dieter Hogrefe, Universität Göttingen

Eike Jessen, Technische Universität München

Ulrich Lang, Universität zu Köln

Paul Müller (Co-Chair), Technische Universität Kaiserslautern

Bernhard Neumair (Co-Chair), Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen

Gerhard Peter, Fachhochschule Heilbronn

Christa Radloff, Universität Rostock

Erwin P. Rathgeb, Universität Duisburg-Essen

Helmut Reiser, LRZ München

Peter Schirmbacher, Humboldt-Universität, Berlin

Uwe Schwiegelshohn, Universität Dortmund

Manfred Seedig, Universität Kassel

Rene Wies, BMW Group

Inhaltsverzeichnis

Netztechnologien

Messen, Analysieren und Überwachen im Rechenzentrum

Technologien im Rechenzentrum

Identity Management

Benutzerzentrierte Lokalisierung für den Einsatz in Shibboleth-basierten Föderationen

Sebastian Rieger

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen (GWDG)
Am Fassberg
37075 Göttingen
sebastian.rieger@gwdg.de

Abstract: Benutzer müssen durch die stetig wachsende Anzahl an Web-Anwendungen (nicht zuletzt durch deren gesteigerte Funktionalität, vgl. „Web 2.0“) zunehmend unterschiedliche Benutzernamen und Passwörter verwalten. In der Vergangenheit haben sich für die Vereinheitlichung der dezentralen Authentifizierung und Autorisierung an Web-Anwendungen Föderationen (basierend auf SAML) und benutzer-zentrierte Verfahren (z.B. OpenID) etabliert. Während letztere für Betreiber und Benutzer einfacher zu verwenden sind, haben SAML-basierte Verfahren eine deutlich höhere Verbreitung insbesondere im wissenschaftlichen Umfeld. Das vorliegende Paper beschreibt eine Erweiterung der SAML-basierten Shibboleth Lösung um eine benutzer-zentrierte Lokalisierung in heterogenen IT-Strukturen über mehrere Föderationen hinweg.

1 Dezentrales Identity Management für heterogene IT-Strukturen

Getrieben durch Entwicklungen wie Asynchronous JavaScript and XML (kurz: AJAX) bzw. „Web 2.0“ sind in den letzten Jahren Web-Anwendungen entstanden, die durch die Verschmelzung von Client- und Server-seitiger Dynamik eine weitaus höhere Interaktivität erlauben [Gar05]. Anwendungsbereiche, die zuvor Desktop-Applikationen vorbehalten waren, können nun dezentral über das World Wide Web zur Verfügung gestellt werden. Durch diesen Trend hat sich auch die Anzahl der verfügbaren Web-Anwendungen erhöht. Web-Anwendungen erlauben ein hohes Maß an Dezentralität z.B. für den gemeinsamen Zugriff auf Inhalte über weltweit verteilte Nutzergruppen hinweg. Um die im Web verarbeiteten Daten zu schützen, erfordern die Anwendungen eine erfolgreiche Authentifizierung und Autorisierung der Anwender z.B. über Benutzernamen und Passwörter. Sichere Single Sign-On Lösungen, die dem Benutzer nach einmaliger Anmeldung Zugang zu unterschiedlichen Web-Anwendungen ermöglichen, können durch unterschiedliche Verfahren realisiert werden. Neben spezialisierten Lösungen, die direkt HTTP-Mechanismen wie z.B. Cookies verwenden, existieren in Föderations-basierten (federated identity) und benutzerzentrierten Verfahren (user-centric) offene Standards (vgl. [RiHi08]). Beide erlauben eine dezentrale Authentifizierung und Autorisierung bzw. eine dezentrale Verwaltung und Speicherung der Identitäten.

Benutzerzentrierte Verfahren wie z.B. OpenID erlauben es den Benutzern, selbst zu entscheiden, für welche Web-Anwendungen sie ihre Identität verwenden. Außerdem ermöglichen sie den Benutzern, im Gegensatz zu föderationsbasierten Verfahren, weltweit eindeutige einheitliche Benutzernamen (z.B. E-Mail Adressen oder URLs). Obwohl benutzerzentrierte Verfahren einige weitere Vorteile gegenüber föderativer Authentifizierung bieten, haben letztere u.a. aufgrund des Security Assertion Markup Language (SAML) Standards insbesondere im wissenschaftlichen Umfeld eine größere Verbreitung. Beispielsweise existiert mit der DFN-AAI des DFN-Vereins eine Föderation für deutsche Forschungseinrichtungen, die das SAML-basierte Verfahren Shibboleth einsetzt [DFAAI].

Dieses Paper beschreibt eine Erweiterung für Shibboleth Identity Provider (IdP), die Vorteile von benutzerzentrierten Verfahren für die föderations-übergreifende Authentifizierung verwendet. Die Implementierung erfolgte für die Integration der Föderation der Max-Planck-Gesellschaft (MPG-AAI) in der Föderation des DFN-Vereins (DFN-AAI). Da die 80 Institute der Max-Planck-Gesellschaft (MPG) ihre IT eigenständig verwalten, unterstützt die Erweiterung neben Shibboleth auch andere Authentifizierungsverfahren für den Zugriff auf die Ressourcen in unterschiedlichen Föderationen. Die Mehrzahl der Institute verwendet LDAP-basierte Verzeichnisse für die Verwaltung ihrer Benutzer. Einzelne Institute verwenden Kerberos oder Datenbanken. Einige Institute betreiben bereits eigene Shibboleth Identity Provider für die Authentifizierung und Autorisierung. Auf der anderen Seite gibt es auch kleinere Institute, die keine eigene Benutzerverwaltung durchführen, bzw. die Verwaltung an externe Dienstleister auslagern.

Aufgrund der Funktion als Vermittler zwischen mehreren Föderationen und verschiedenen Authentifizierungsverfahren wird die in diesem Papier vorgestellte Erweiterung als „IdP Proxy“ bezeichnet. Der IdP Proxy ermöglicht es die Eigenständigkeit der Max-Planck-Institute hinsichtlich des Identity Managements aufrecht zu erhalten, ohne alle 80 Institute separat in externe Föderationen (z.B. der DFN-AAI) integrieren zu müssen.

1.1 Föderatives Identity Management in wissenschaftlichen IT-Strukturen

Um die Interoperabilität von Web-basierten Single Sign-On Verfahren unterschiedlicher Hersteller zu gewährleisten, wurde von der OASIS die Security Assertion Markup Language (SAML) als einheitlicher Standard verabschiedet [SAML]. SAML liegt aktuell in der Version 2.0 vor. Bei der Authentifizierung und Autorisierung wird in SAML zwischen dem Service Provider (SP), der die zugriffsbeschränkte Ressource vorhält, und dem Identity Provider (IdP), der berechtigte Identitäten (Benutzernamen, Passwort) bereitstellt, unterschieden. Ein Service Provider kann beispielsweise bei einem Verlag bzw. einem externen Dienstleister implementiert werden. Identity Provider werden dezentral in den Instituten (als „Heimat-Organisationen“) realisiert und ermöglichen die Authentifizierung und Autorisierung der Instituts-Benutzer in SAML-basierten Föderationen. Eine Föderation umfasst mehrere SPs und IdPs, die sich untereinander vertrauen. Dies ermöglicht des über die IdPs angebundenen Benutzern ein Single Sign-On an den von den SPs zur Verfügung gestellten Ressourcen.

Hierfür stellt der IdP nach einer erfolgreichen Authentifizierung ein digital signiertes Token (Assertion) für den Benutzer aus, dessen Signatur durch die SPs überprüft wird. Die Vertrauensstellung wird dabei über X.509 Zertifikate gewährleistet, wie sie z.B. die DFN-PKI [DFPKI] anbietet. Zertifikate und Dienstzugriffspunkte werden in den sog. Metadaten verwaltet, die die Föderation definieren. Für die Autorisierung werden Attribute verwendet, die der IdP zusätzlich zur Assertion bereitstellt und die von den SPs verwendet werden.

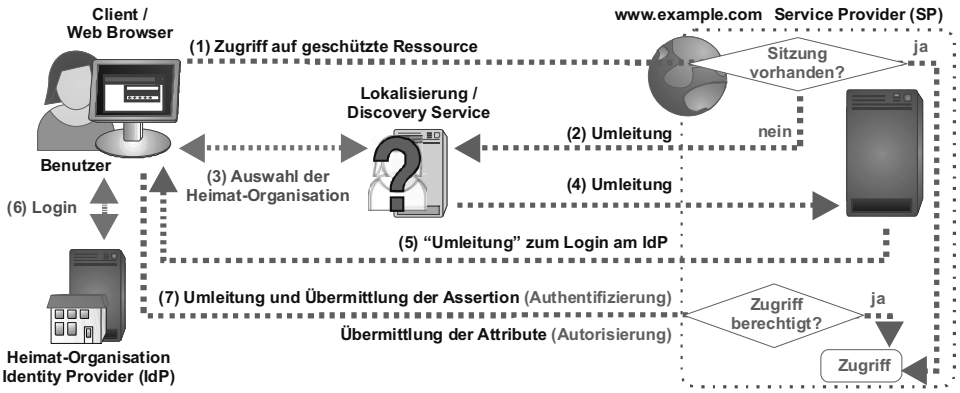


Abbildung 1: Föderatives Identity Management am Beispiel von Shibboleth 2.0.

In wissenschaftlichen IT-Strukturen hat sich das SAML-basierte Shibboleth Verfahren etabliert, das auch bei der DFN-AAI verwendet wird. Abbildung 1 zeigt den Ablauf einer Authentifizierung und Autorisierung bei Shibboleth in der Version 2.0. Für die Lokalisierung des für den Benutzer zuständigen IdPs verwendet Shibboleth einen sog. Discovery Service (DS). Der Benutzer wählt hier die Heimat-Organisation, der er angehört, aus. Ab Shibboleth 2.0 wird auch eine passive Auswahl des zuständigen IdP durch den SP unterstützt. Hierbei erfolgt die Lokalisierung passiv auf Basis der IP Adresse des Clients. Nach dem erfolgreichen Login am IdP übermittelt der Benutzer die von dem IdP ausgestellte Assertion an den SP, der diese prüft und anhand der übermittelten Attribute die Autorisierung durchführt. Konnte die Authentizität des Benutzers erfolgreich geprüft werden und ist dieser anhand der Autorisierung berechtigt, wird der Zugriff auf die Ressource gewährt. Eine Referenz auf die Assertion wird als Cookie für den IdP im Web-Browser gespeichert, so dass ein nachfolgender Zugriff auf eine Ressource eines anderen SPs innerhalb der Föderation keine erneute Anmeldung erfordert. Zusammen mit den am SP aufgebauten Sitzungen (HTTP sessions) wird so ein Single Sign-On innerhalb der Föderation erzielt.

Föderative Authentifizierungsverfahren (insbesondere Shibboleth) sind derzeit sowohl in wissenschaftlichen als auch in wirtschaftlichen IT-Strukturen weit verbreitet. Durch die Interoperabilität des SAML Standards bieten auch viele Software-Hersteller Lösungen für föderative Authentifizierung an (z.B. Microsoft ADFS, Sun OpenSSO oder Novell Access Manager). Andererseits sind die konkreten Implementierungen (z.B. Shibboleth) häufig vergleichsweise komplex zu administrieren.

Für den Benutzer wird zusätzlich die Verwendung komplex, sofern er Ressourcen unterschiedlicher Föderationen verwenden kann. Für jede Föderation muss der Benutzer seine Zugangsdaten verwalten, sowie seinen IdP bzw. seine Heimat-Organisation und zugehörige Föderationen kennen. Die Lokalisierung wird dabei für den Benutzer zusätzlich erschwert, wenn mehrere Föderationen zu einer Konföderation zusammengefasst werden (vgl. eduGAIN [EGAIN]). Er muss in diesem Fall zwei DS Instanzen durchlaufen, und zunächst seine Heimat-Föderation und dann seine Heimat-Organisation auswählen. Neben dem Mehraufwand durch die zusätzliche Auswahl bedingt dies, dass der Benutzer seine Zugehörigkeit zu einer Heimat-Föderation (z.B. DFN-AAI, SWITCH-AAI) überhaupt kennt bzw. namentlich zuordnen kann.

1.2 Benutzerzentrierte Ansätze für das Identity Management

Während die Lokalisierung bei föderativen Authentifizierungsverfahren auf einem zentralen DS basiert, erfolgt sie bei benutzerzentrierten Verfahren dezentral anhand eines vom Benutzer ausgewählten eindeutigen Identifikationsmerkmals. Der Benutzer wählt beispielsweise eine Karte (I-Card), die seine Zugehörigkeit zu einem Provider, der eine ähnliche Funktion wie der Identity Provider in Föderationen realisiert, angibt. Zusätzlich hat er die Möglichkeit auszuwählen, welche Informationen er an den Consumer, der eine ähnliche Funktion wie der Service Provider in Föderationen aufweist, übermitteln möchte. Dies erhöht neben der Usability hinsichtlich der Lokalisierung auch den Datenschutz. Nicht zuletzt durch die vereinfachte Lokalisierung sind Implementierungen benutzerzentrieter Authentifizierungsverfahren in der Regel weniger komplex als Föderationsbasierte. I-Cards werden jedoch noch nicht von allen Web-Browsern für die Authentifizierung an Web-Anwendungen unterstützt. Eine häufig verwendete Alternative für benutzerzentrierte Verfahren bilden daher URLs oder E-Mail-Adressen als Identifizierungsmerkmal der Benutzer. URLs und E-Mail-Adressen werden beispielsweise von OpenID verwendet, das nicht zuletzt aufgrund der Unterstützung durch Google und Yahoo immer mehr an Bedeutung gewinnt.

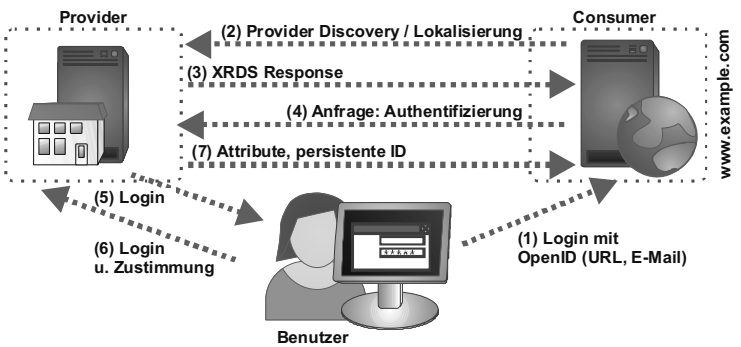


Abbildung 2: Benutzerzentriertes Identity Management mit OpenID.

Alternativen zu OpenID bilden z.B. Microsoft CardSpace oder Higgins. Andere benutzerzentrierte Authentifizierungsverfahren integrieren OpenID (z.B. sxip, OAuth). Zum aktuellen OpenID Standard [OID] in der Version 2.0 existieren einige Erweiterungen, die beispielsweise neben dem Single Sign-On auch die Übertragung von Attributen für die Autorisierung, analog zu den föderativen Authentifizierungsverfahren, erlauben. Ein Beispiel für eine Authentifizierung und Autorisierung mittels OpenID zeigt die Abbildung 2. Der Benutzer meldet sich am Consumer mit seiner OpenID an. Anhand der E-Mail Adresse oder einem Zugriff auf die URL ermittelt der Consumer den zuständigen Provider, an den er anschließend eine Authentifizierungsanfrage richtet. Die eigentliche Authentifizierung erfolgt vom Benutzer über das Login am Provider. Ist die Authentifizierung erfolgreich, so leitet der Provider den Benutzer wieder an den Consumer. Hierbei kann er auch Attribute für die Autorisierung des Benutzers am Consumer übermitteln. Die Autorisierung erfolgt durch die Überprüfung der erforderlichen Attribute am Consumer. Durch die Realisierung einer Sitzung (Cookie) am Provider wird ein Single Sign-On über unterschiedliche Consumer realisiert.

Benutzerzentrierte Verfahren wie z.B. OpenID sind vergleichsweise neu und weisen daher noch einige Sicherheitslücken auf. Beispielsweise wurden einige typische Angriffe auf Web-Anwendungen wie Phishing, Replay Attacken und Implementierungsfehler (z.B. Cross-Site Scripting und Cross-Site Request Forgeries) in [Tsyr07] vorgestellt.

Obwohl für benutzerzentrierte Verfahren kein einheitlicher Standard existiert, hat das OpenID Verfahren seit 2008 eine große Verbreitung erlangt. Dies resultiert vor allem aus der Unterstützung von OpenID durch Google und Yahoo. Allerdings betreiben diese bislang nur OpenID Provider und keine Consumer für ihre eigenen Dienste. Sie vergeben somit lediglich OpenIDs für ihre Benutzer. Benutzer mit einer OpenID (z.B. von Yahoo) können jedoch nicht auf die Dienste von Google oder Yahoo zugreifen. Auf diese Weise sind Benutzer gezwungen nach wie vor separate Accounts bei Google und Yahoo zu registieren. OpenID Consumer werden momentan eher von kleineren Anbietern betrieben, die sich dadurch eine Steigerung der Anzahl ihrer Benutzer erhoffen. Solange große Unternehmen ausschließlich OpenID Provider anbieten, werden auch zukünftig unterschiedliche Benutzernamen bzw. OpenIDs auf der Seite der Benutzer erforderlich sein. Google verwendet zusätzlich Erweiterungen für den OpenID Standard, um z.B. die Anmeldung mittels E-Mail-Adresse zu ermöglichen. Während dies die Usability erhöht, könnte diese Abweichung vom Standard (in der Regel verwenden OpenID Provider URLs) zukünftig auch der Interoperabilität von OpenID schaden und so die Bindung der Kunden an die von Google unterstützten Dienste erhöhen.

Die Bindung der Benutzer an ihren OpenID Provider ist mitunter ein allgemeiner Nachteil von OpenID. Hat ein Benutzer seine OpenID für unterschiedliche Dienste registriert, so ist er abhängig von der Verfügbarkeit der OpenID durch seinen Provider. Fällt der Dienst des OpenID Providers aus oder ändert dieser seine Geschäftsbedingungen, und der Benutzer wechselt den Anbieter, so sind unter Umständen auch die Consumer und dort hinterlegte Daten, für die er seine OpenID verwendet hat, nicht mehr zugänglich.

2 Benutzerzentrierte Lokalisierung mit Shibboleth

Obwohl benutzerzentriertes Identity Management, wie im vorherigen Abschnitt geschildert Vorteile in Bezug auf die geringere Komplexität, einheitliche Lokalisierung und Usability aufweisen, sind SAML-basierte Föderationen derzeit insbesondere in wissenschaftlichen IT-Strukturen das Standard-Verfahren (vgl. z.B. Shibboleth innerhalb der DFN-AAI [DFAAI] in Deutschland, InCommon [INCO] in den USA oder simpleSAMLphp für die FEIDE [FEIDE] Föderation in Skandinavien) für dezentrales Identity Management.

Darüber hinaus existieren neben freien Implementierungen auch kommerzielle Lösungen z.B. in Microsoft ADFS oder Sun OpenSSO. Der SAML Standard ist weitgehend ausgereift und es liegen derzeit keine Sicherheitslücken, wie für OpenID geschildert, vor. Außerdem können Betreiber ihre IdPs in unterschiedliche Föderationen integrieren und so die Abhängigkeit von einem konkreten Provider, wie im vorherigen Abschnitt beschrieben, vermeiden.

Um einige Vorteile benutzerzentrierter Verfahren für föderatives Identity Management nutzbar zu machen, wurden im Rahmen der Realisierung der Föderation der Max-Planck-Gesellschaft (MPG-AAI [MPAAI]) einige Erweiterungen für Shibboleth realisiert, die in den folgenden Abschnitten beschrieben werden.

2.1 Shibboleth IdP Proxy für heterogene IT-Strukturen

Innerhalb der Max-Planck-Gesellschaft bestand die Anforderung Shibboleth für die Authentifizierung und Autorisierung verteilter Forschungsgruppen einzusetzen. Außerdem sollten die ca. 80 Institute Zugriff auf externe Dienstleister bzw. Verlage erhalten, die ihre Authentifizierung und Autorisierung bereits auf Shibboleth umgestellt haben. Die Mehrheit dieser externen Service Provider ist bereits in der ebenfalls Shibboleth-basierten DFN-AAI Föderation des DFN-Vereins integriert. Um die Dienste innerhalb der DFN-AAI für die Benutzer der Max-Planck Institute nutzbar zu machen, wurde eine Anbindung der MPG-AAI an die DFN-AAI angestrebt.

Eine einfache Lösung für die Integration der eigenständigen Institute in die DFN-AAI bildet die Registrierung separater IdPs für jedes Institut der MPG innerhalb der DFN-AAI. Dies hätte jedoch zur Folge gehabt, dass jedes Institut auch einen IdP installieren und warten muss. Aus Sicht des DFN-Vereins war es außerdem nicht wünschenswert, alle 80 Institute der Max-Planck-Gesellschaft separat zu integrieren, da dies die Verwaltung des DS erschwert hätte. Insbesondere wäre hierbei die Verwendung der DFN-AAI aufgrund der Länge der Liste der Heimat-Organisationen (Universitäten, Forschungseinrichtungen) durch die zusätzlichen MPG Institute schwieriger geworden. Benutzer anderer Einrichtungen hätten bei der Auswahl ihrer Heimat-Organisation am DS der DFN-AAI jeweils die 80 Institute der MPG durchgehen müssen. Die MPG-AAI sollte zusätzlich offen für die Integration weiterer Föderationen, z.B. von internationalen Forschungsgruppen oder -netzen, konzipiert werden. Dies schließt auch die eigenständige Integration einzelner Institute in weitere Föderationen mit ein.

Als Alternative zur Registrierung separater IdPs für die Institute wurde daher ein einzelner IdP entwickelt, der als Proxy die gesamte MPG in der DFN-AAI repräsentiert. Dieser „IdP Proxy“ ist dabei sowohl in der DFN-AAI als auch in der MPG-AAI registriert, wie in Abbildung 3 illustriert. Verwenden Benutzer eines Instituts der MPG einen Dienst innerhalb der DFN-AAI, so wählen sie im DS des DFN-Vereins „Max-Planck-Gesellschaft“ aus, und werden zum Login an dem IdP Proxy weitergeleitet. An diesem können sich die Benutzer mit dem Account ihres lokalen Instituts anmelden.

Um die heterogene Struktur der Benutzerverwaltung an den Instituten abzubilden, können Institute bezüglich der Anbindung an den IdP Proxy zwischen drei Optionen wählen, die in Abbildung 3 gezeigt werden.

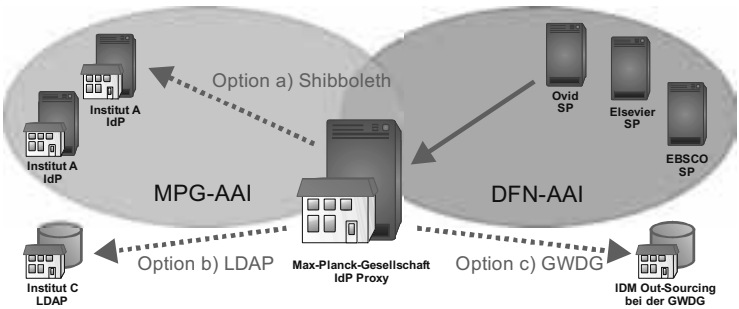


Abbildung 3: Shibboleth IdP Proxy als Bindeglied zwischen heterogenen IT-Strukturen.

Bei dem in Option a) gezeigten Betriebsmodell betreibt das Institut selbst einen Shibboleth IdP innerhalb der MPG-AAI, der dann vom IdP Proxy für die Authentifizierung und Autorisierung verwendet wird. Besitzt das Institut keine eigenen Kapazitäten für den Betrieb eines Shibboleth IdP, so kann auch eine andere bestehende Benutzerverwaltung angebunden werden. Beispielsweise könnte der IdP Proxy gemäß Option b) eine Datenbank, ein Kerberos KDC oder, wie in der MPG am häufigsten verwendet, einen LDAP Server des Instituts abfragen. Innerhalb der MPG existieren darüber hinaus kleinere Institute, die keine eigene Benutzerverwaltung durchführen bzw. diese an externe Dienstleister auslagern. Die Option c) in Abbildung 3 zeigt diese Auslagerung des Identity Mangements an die Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen (GWDG) als externen Dienstleister. Der IdP Proxy kann zusätzlich zur in Abbildung 3 gezeigten MPG-AAI und DFN-AAI auch in weitere Föderationen aufgenommen werden. Beispielsweise können Föderationen anderer Länder oder Forschungsgruppen am IdP Proxy angebunden werden. Dies unterstützt auch die Fluktuation von Personen und Projekten (z.B. temporäre Aufnahme von Gastforschern etc.). Der IdP Proxy bildet somit eine zentrale Instanz, um dezentrale Authentifizierung und Vertrauensstellungen zu realisieren.

Abbildung 4 zeigt die Implementierung des IdP Proxy als Erweiterung für einen Shibboleth 2.0 IdP. Der Shibboleth IdP ist eine Web-Anwendung die in der Regel in einem Apache Tomcat Application Server läuft. Kern des IdP Proxy ist das Shib Proxy Servlet, das als Login Handler (Auth Engine) in dem Shibboleth IdP konfiguriert wird.

Sofern für Benutzer, die durch einen DS an den IdP Proxy weitergeleitet werden, noch keine Sitzung (basierend auf einem vom Web Browser übermittelten Cookie) am Proxy existiert, wird eine Login-Seite angezeigt. Andernfalls wird vom SSO Profile Modul direkt eine Assertion erstellt, signiert und an den SP übermittelt. Die Login-Seite ist eine Java Server Page, die den Benutzernamen (E-Mail Adresse) und das Passwort an das Shib Proxy Servlet übermittelt. Anhand der Domain der E-Mail Adresse (bzw. dem Realm) ermittelt der IdP Proxy den zuständigen IdP. Die Zuordnung zu den entsprechenden Instituten wird in der Konfiguration des IdP Proxy definiert.

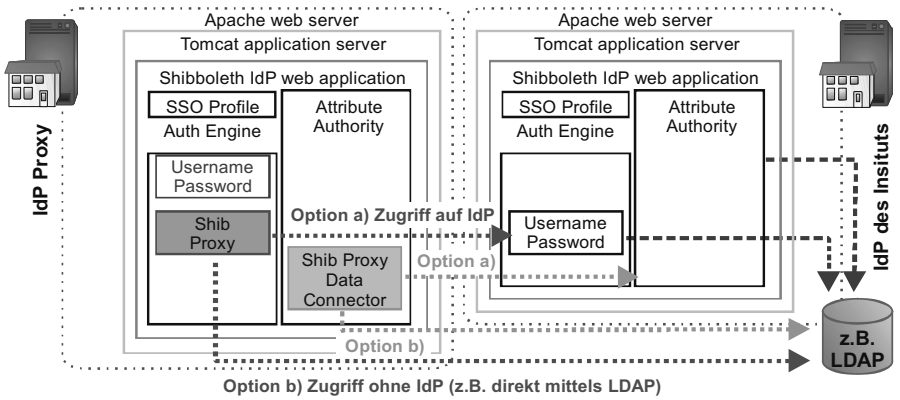


Abbildung 4: Implementierung des IdP Proxy als Erweiterung für den Shibboleth IdP

In Abbildung 4 werden zwei Optionen für die Anbindung eines Instituts an den IdP Proxy beschrieben, die auch in Abbildung 3 gezeigt wurden. Option a) zeigt den Fall, in dem das externe Institut einen eigenen Shibboleth IdP betreibt. In diesem Fall wird die Authentifizierung und Autorisierung direkt an den IdP des Instituts weitergeleitet. Hierbei werden vom Proxy unterschiedliche, von Shibboleth angebotene, Login Verfahren unterstützt (z.B. HTTP- oder Formular-basierte Authentifizierung jeweils für Shibboleth 1.3 und 2.0). Bei der Formular-basierten Authentifizierung fügt der IdP Proxy die auf seiner Login-Seite eingegebene Kennung (der Realm am Benutzernamen kann optional zuvor gefiltert werden) im Formular des IdPs am Institut ein und schickt dieses ab.

Sendet der IdP des Instituts daraufhin eine Assertion (SAML Response) zurück, so wird die enthaltene digitale Signatur anhand der X.509 Zertifikate in den Metadaten der Föderation bzw. der Konfiguration des IdP Proxy überprüft. Sofern die Überprüfung und damit die Authentifizierung erfolgreich war, erstellt der IdP Proxy seinerseits eine Assertion, die er dann z.B. an den externen SP in einer anderen Föderation sendet. Überprüfung und Erzeugung der Assertion im IdP Proxy verwenden analog zum Shibboleth IdP OpenSAML [OSAM]. Der Name Identifier der Shibboleth Sitzung, wird aus der SAML Response extrahiert und im Proxy zwischengespeichert. Dies ermöglicht es, nachfolgende Autorisierungs- bzw. Attributanfragen (wie bei Shibboleth 1.3 verwendet), dem Benutzer zuzuordnen, für den die Assertion erstellt wurde. Dadurch kann das Institut bzw. der IdP ermittelt werden, von dem die Attribute für den Benutzer abgerufen werden müssen. Um die Attribute des Benutzers zu ermitteln, wurde zusätzlich zum Shib Proxy Servlet ein Custom Data Connector für den Shibboleth IdP implementiert.

Scoped Attribute (z.B. „mmuster@institut-a.mpg.de“) werden „prescoped“ verwendet, um die in den Instituten vergeben Scopes im Proxy beibehalten zu können. SAML Attribute Statements an Shibboleth 2.0 IdPs werden digital signiert und geprüft. Verfügt das Institut dem der Benutzer angehört über keinen eigenen IdP, so kann analog zu Option b) und c) aus Abbildung 3, die Authentifizierung gegen ein separates System (z.B. Datenbank, Verzeichnisdienst) erfolgen. Abbildung 4 zeigt hierfür die Option b).

Das Shib Proxy Servlet sowie der zugehörige Data Connector können z.B. JNDI verwenden, um mittels LDAP(S) die Authentifizierung durchzuführen und Attribute des Benutzers zu ermitteln. Alternativ können Datenbanken (über JDBC) oder separate Authentifizierungssysteme (z.B. Kerberos) über JAAS angebunden werden.

Unabhängig von dem durch das Institut gewählten Betriebsmodell können die für den Benutzer ermittelten Attribute vor der Übermittlung an den SP gefiltert oder modifiziert werden. Beispielsweise können Scopes auf „@mpg.de“ geändert oder Attribute, deren Inhalt nur innerhalb der MPG übertragen werden darf, gefiltert werden. Dadurch wird der Datenschutz bei der Übermittlung von Attributen zwischen unterschiedlichen Föderationen gewährleistet. Um den Benutzern vergleichbar mit den benutzerzentrierten Verfahren, wie in Abschnitt 1.2 beschrieben, die Möglichkeit zu geben, die übermittelten Attribute selbst zu kontrollieren, wurde auf dem IdP Proxy der in der Schweiz für die SWITCH-AAI entwickelte ArpViewer [ArpVie] installiert. Der ArpViewer zeigt dem Benutzer alle ermittelten Attribute an und erfordert dessen explizite Zustimmung vor der Übermittlung.

2.2 Integration mehrerer Föderationen ohne zusätzlichen Lokalisierungsdienst

Wie im vorherigen Abschnitt beschrieben, erlaubt der IdP Proxy den Benutzern der MPG Institute Zugriff auf Ressourcen in unterschiedlichen Föderationen zu nehmen. Um hierbei die Vorteile der Lokalisierung anhand globaler Benutzernamen, wie im Abschnitt 1.2 für benutzerzentrierte Verfahren beschrieben, zu nutzen, verwenden die Benutzer für die Anmeldung am IdP Proxy ihre E-Mail Adresse. Anhand der Domain der E-Mail Adresse kann der Proxy das für den Benutzer zuständige Institut sowie das gewünschte Authentifizierungs- und Autorisierungsverfahren ermitteln. Die Benutzer müssen sich daher nur ihre E-Mail Adresse, nicht die Zugehörigkeit zu Institutionen oder unterschiedlichen Föderationen, merken. Anders als bei alternativen Verfahren wie z.B. eduGAIN [EGAIN], die ebenfalls mehrere Föderationen miteinander verbinden, erfolgt für den Benutzer nur eine Lokalisierung. Er muss nur die Max-Planck-Gesellschaft und nicht anschließend in einem weiteren DS das zuständige Institut auswählen. Außerdem kann in den externen Föderationen ein einziger IdP für die gesamte MPG angegeben werden und so die Übersichtlichkeit des DS für den Benutzer gewährleistet werden. E-Mail Adressen werden nur für das Login am IdP Proxy verwendet. Sie werden nicht an die Service Provider übermittelt. Dem SP ist nur eine Shibboleth Sitzung bzw. deren Name Identifier bekannt, über den er z.B. Attribute für den Benutzer anfordern kann. Zusätzlich können übermittelte Attribute auch, wie im vorherigen Abschnitt beschrieben, gefiltert werden.

3 Fazit und Ausblick

Föderative Verfahren für die dezentrale Authentifizierung und Autorisierung an Web-Anwendungen sind insbesondere in wissenschaftlichen IT-Strukturen nach wie vor der Standard. Eine Integration mehrerer Föderationen stellt Herausforderungen für die Usability und den Datenschutz bezüglich der übermittelten Attribute dar.

Der in diesem Paper vorgestellte IdP Proxy adressiert diese Anforderungen für die Institute der MPG. Dabei werden Eigenschaften von benutzerzentrierten Verfahren (z.B. globaler Benutzername in unterschiedlichen Föderationen, einfache Lokalisierung und benutzerzentrierter Datenschutz) in der skizzierten Shibboleth Erweiterung implementiert. Eine Kaskadierung der DS bei der Verbindung mehrerer Föderationen wird hierdurch vermieden. Die Integration mehrerer Föderationen wird auch von anderen Lösungen wie eduGAIN [EGAIN] sowie benutzerzentrierten Verfahren adressiert. Zukünftig sind darauf basierende Erweiterungen für den IdP Proxy realisierbar.

Aktuell befindet sich eine Integration von OpenID im IdP Proxy in der Entwicklung. Benutzer der MPG erhalten dabei eine OpenID (z.B. nach dem Muster https://shib-idp.mpg.de/id/mmuster@inst-a.mpg.de) am IdP Proxy. Auch eine Anmeldung über OpenID auf der Login-Seite des IdP Proxy wäre möglich. Dies würde allerdings eine Konvertierung der übermittelten Attribute erfordern. Momentan bieten die innerhalb der MPG verwendeten Web-Dienstleister noch keine OpenID Consumer an, so dass die Unterstützung von OpenID keine explizite Anforderung darstellt.

Literaturverzeichnis

[ArpVie] SWITCH: uApprove ArpViewer, http://www.switch.ch/aai/support/tools/arpviewer.html, abgerufen am: 12.1.2009.

[DFAAI] DFN: DFN-AAI Einfacher Zugang zu geschützten Ressourcen, http://www.dfn.de/index.php?L=0&id=75522, abgerufen am: 18.1.2008.

[DFPKI] DFN: DFN-PKI Überblick, http://www.pki.dfn.de/, abgerufen am: 18.1.2008.

[EGAIN] eduGAIN: http://www.edugain.org, abgerufen am: 12.1.2009.

[FEIDE] UNINETT: Feide, http://feide.no, abgerufen am: 12.1.2009.

[Gar05] Garret, J. J.: Ajax: A New Approach to Web Applications, http://www.adaptivepath.com/ideas/essays/archives/000385.php, abgerufen am: 12.1.2009.

[INCO] Internet2: InCommon, http://www.incommonfederation.org, abgerufen am: 12.1.2009.

[MPAAI] MPG: MPG-AAI, https://aai.mpg.de, abgerufen am: 12.1.2009.

[OID] OpenID: Specifications, http://openid.net/developers/specs, abgerufen am: 12.1.2009.

[OSAM] Internet2: OpenSAML, www.opensaml.org, abgerufen am: 12.1.2009.

[RiHi08] Rieger S.; Hindermann T.: Dezentrales Identity Management für Web- und Desktop-Anwendungen. In (Müller, P.; Neumair, B.; Dreo Rodosek, G., Hrsg.): Proc. 1. DFN-Forum Kommunikationstechnologien, Kaiserslautern 2008. Gesellschaft für Informatik , Bonn, 2008; S. 107-116.

[SAML] OASIS: Security Services (SAML) TC, http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security, abgerufen am: 12.1.2009.

[Tsyr07] E. Tsyrklevich. V. Tsyrklevich: OpenID - Single Sign-On for the Internet (Blackhat USA, 2007), https://www.blackhat.com/presentations/bh-usa-07/Tsyrklevich/Whitepaper/bh-usa-07-tsyrklevich-WP.pdf, abgerufen am: 12.1.2009.

MetaVoIP – Sharing Contact Information over Organizational Boundaries

Frank Eyermann, Iris Hochstatter

Institut für Technische Informatik
Universität der Bundeswehr München
Werner-Heisenberg-Weg 39
85577 München
Frank.Eyermann@unibw.de
Iris.Hochstatter@unibw.de

Abstract: Nowadays, many projects are carried out in a collaboration of people or groups from different institutions and/or enterprises. Such a *virtual organization* is characterized by high communication needs but does not operate in one single environment. MetaVoIP eases the communication as it automatically combines contact information from different PBXs and provides it to all partners. For a given virtual organization the MetaVoIP server is periodically retrieving contact information from the participating organizations via different data source drivers. The PBX connectors provide the telephony functionality independent of an organization's PBX, and the user GUI allows for easy user management as well as click-to-call. MetaVoIP has been implemented and tested; drivers exist for the Asterisk PBX and data input from LDAP and Asterisk configuration files.

1 Introduction

A cooperation of different enterprises, institutions or generally speaking organizations always increases the communication needs between those organizations. Nowadays, communication is already greatly support by electronic media, such as email, wikis or special collaboration tools. Yet, voice communication is still the most important means of communication. Making a phone call, however, requires knowing the phone number of the organization and the extension of the callee. Within an organization a corporate directory typically satisfies this information need. Because of privacy and security issues these directories are mostly not publicly readable, leaving project partners calling a receptionist and being connected. Some projects might run project-specific directories, but those are usually not linked with the corporate one and therefore have to be updated manually whenever a change occurs. MetaVoIP closes this gap. The name MetaVoIP is modeled after the term metadirectory being a directory service, which combines the data from various directories and displays them in a uniform way [Je07]. MetaVoIP combines data from different private branch exchanges (PBXs) and provides a centralized phone book with additional user services. The central phone book is automatically built on the configurations of the local phone systems. The phone systems can be Voice-over-IP-based, but this is not a requirement.

The remainder of this paper is organized as follows: Section 2 discusses related work for the MetaVoIP functionality. We then follow up with the design of MetaVoIP, which includes the requirements, MetaVoIP topology and architecture. The section closes with in-depth details about the different abstraction layers and roles important within MetaVoIP. Section 4 gives implementation details about all components of MetaVoIP including security issues. We conclude the paper in Section 5 and give an outlook to future work.

2 Related Work

MetaVoIP combines contact information for virtual organizations and keeps them up-to-date automatically. To our knowledge, no other approach is a solution to this particular problem. Still, the research areas of (meta-)directories and approaches to phone book applications are highly related and we have investigated them closely.

2.1 Directories and Metadirectories

A directory service is a network service providing information about objects, e.g. a directory service implementing an address book provides information about people. The Lightweight Directory Access Protocol (LDAP) [LD97] allows querying and modifying directory services and organizes the data in a tree. LDAP is the replacement for X.500 and was specifically designed to be less complex and extensive. The data-model is object-oriented and the directory tree can be distributed over several servers.

A metadirectory service collects and integrates data from different directory services or databases. MetaVoIP is designed to manage contact information from various PBXs where its organizations form a kind of virtual organization. Examples include research institutions collaborating in a joint project or companies and their subcontractors. Metadirectories as part of identity management solutions exist for a variety of applications, both commercial [Ev08] [Mi08] [Ra08] [Or08] and open-source. MetaComm [Fr00] was an approach by Bell Labs researchers towards a metadirectory for telecommunications in 2000. It combined the metadirectory with the functionality to modify the data in the metadirectory and ensured data integrity at the same time. DS4J also offers metadirectory functionality and is implemented in JBoss [DS08]. [Ma03] describes a project that is related to the OpenLDAP community and presents a metadirectory gathering the data on a regular basis from heterogeneous sources.

2.2 Asterisk

Asterisk [Va05] is an open-source PBX including all common telephony functionalities. Originally developed by Digium Inc., today many other developers have joined the project. Asterisk supports plain old telephone service (POTS) and Integrated Services Digital Network (ISDN) as well as VoIP with different protocols. Those features allow for various enhancements and easy extensibility and make Asterisk a very powerful tool.

LDAP support was introduced to Asterisk only in the most current version 1.6. Previous versions, which are still widely deployed, do not support LDAP [Vo08].

A great number of extensions to asterisk destined to administrators as well as users have been developed. An overview with more than 100 entries showing ready-to-use distributions, configuration managers, solutions for service providers, billing and call reporting, call center and contact center management solutions, status viewers and user interfaces can be found in [AG08]. However, none of these tools and extensions addresses the management of a federated environment consisting of several independent organizations and thus independent PBX and directory services.

3 Design of MetaVoIP

Before we describe MetaVoIP in detail, we first describe a typical scenario and derive from this the requirements to a metadirectory approach supporting the transparent sharing of contact information over organizations. Second, the topology of a MetaVoIP virtual organization and architecture of MetaVoIP is described. The MetaVoIP authentication and authorization mechanisms, as well as its roles are given in the third subsection. At the end, the MetaVoIP data model is specified.

3.1 Scenario and Requirements

Within an enterprise or cooperation several subsidiaries or institutions (termed organizations in this paper) are collaborating. A single directory of all participants of all organizations is missing, because of legal or organizational aspects, or because only one branch or department is involved in the cooperation. Yet frequent voice communication between the partners is necessary. Each of the organizations operates its own private branch exchange, which might be based on any telecommunication technology. The PBXs are produced by different manufacturers and administrated independently by each organization. However, each PBX is connected to the network and provides an interface in order to remotely control the PBX.

Although no central directory for all participants exists, each partner itself operates a local directory. The directories may be based on different technologies, e.g., LDAP, relational databases or simple text files and can be accessed over the network. The data in the directory contains at least the persons' telephone numbers. A local directory is termed data source in this paper.

In order to simplify contacting project partner by phone MetaVoIP must show a list of all published users of all organizations. Organizations can limit the number of users published by disallowing MetaVoIP to read parts of the directory. Beneficial for users is the possibility to call another user with one click. Especially in large cooperations users of MetaVoIP can be further assisted with search and filter functionalities. If possible the status of an extension (e.g., offline, online, or busy) should be displayed, too. Because of privacy issues access to MetaVoIP must be authenticated and authorized.

3.2 MetaVoIP Topology

MetaVoIP is designed as a centralized server application. It abstracts from organizations and describes the topology with sites. For simplicity it is assumed that each organization is located at one site. A site is defined as a location with exactly one data source and exactly one PBX. Mappings must be made, if this assumption does not hold. An example with three sites is shown in Figure 1. The sites with each a data source and a PBX are connected to the MetaVoIP server. Users can access the services of MetaVoIP over the network.

3.3 Architecture

The architecture of MetaVoIP consisting of four main building blocks is depicted in Figure 2. An extended three-tier approach is used. The *User GUI* provides a graphical user interface and thus implements the presentation layer. The *Application logic* implements the business logic of MetaVoIP: Firstly, it serves the *User GUI* and supplies it with data from a local database. It performs the user commands, e.g., initiating a call with the help of the PBX abstraction layer and a *PBX connector*. Secondly, it periodically replicates data from the organizations in its own local database. The persistence layer of a three-tier architecture consists in MetaVoIP of three parts: a local database, which stores all data necessary for operation of MetaVoIP, a Data source connecter, and a PBX connector. We will describe the data model of MetaVoIP in Section 3.5, and the two connectors subsequently.

3.3.1 Data source connector

The *data source connector* provides an interface to the user information stored remotely at each site and abstracts from the concrete representation of the data. A more detailed view of the data source connector is shown in Figure 3. Different *data source drivers* may be implemented in order to connect to different types of data sources, e.g., LDAP, X.500, SQL database, or a driver to retrieve the user data directly out of a PBX.

The *data source abstraction layer* provides a uniform interface towards the application logic and chooses the appropriate driver depending on the OrgSite.orgid field (see Section 3.5). As each site operates by definition only one data source one driver is

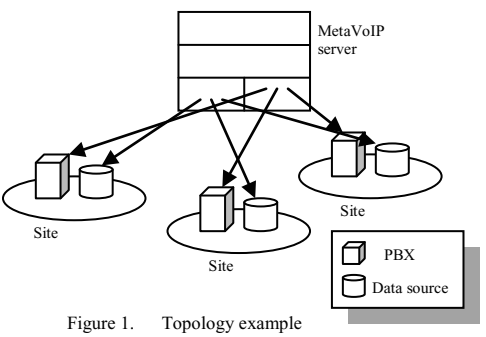


Figure 1. Topology example

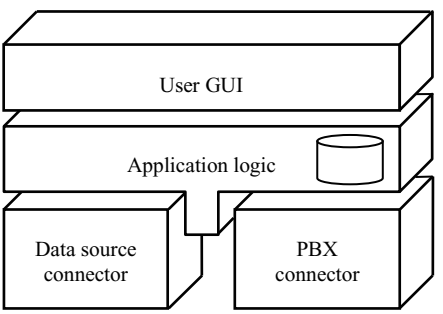


Figure 2. Architecture of MetaVoIP

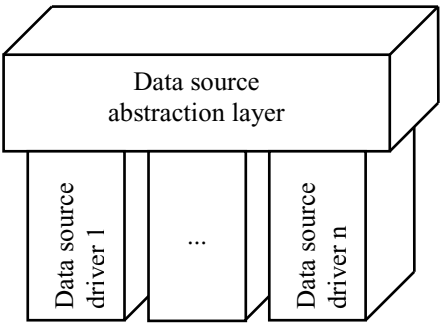


Figure 3. Data source connector

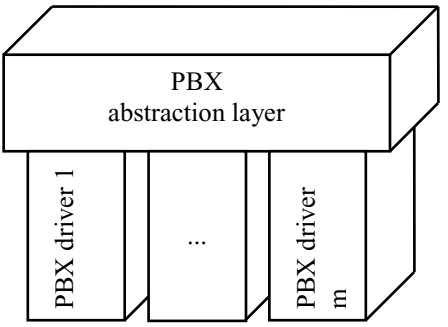


Figure 4. PBX connector

instantiated per site.

A data source driver must implement the function to read user information from the sites. User information consists of the list of all users, their attributes, as well as their phone accounts, i.e., the different telecommunication means with which the user can be reached. User passwords can only be replicated if they are stored as clear text in the data source. Only the data source itself can work with encrypted or hashed password. In this case if a user logs on to MetaVoIP the user is authenticated directly against the data source at his site (see also Section 4.4).

3.3.2 PBX connector

The *PBX connector* is the link between the application logic and the PBX system at a site. Because of the different types of PBX systems and the different access possibilities respectively, a PBX connector implements a driver concept, analogously to the one of the data source connector (see Figure 4).

A PBX driver implements two functions: First it periodically checks the status of a phone account, and second it can establish a phone call. The status of a phone account shows if a phone is currently reachable. This is of special interest for phones, which are not always connected as, e.g., SIP soft phones. In this case showing the connected state of phone accounts to a user will give him an impression, who is online and reachable and who is not. The known states are: up, down, busy, and unknown. The second function is directly derived from the requirement that users should be able to place a call by clicking the callee in the user GUI. The function instructs the PBX to establish a phone call between two or more named phone accounts.

3.4 Roles

MetaVoIP facilitates a role-based authorization scheme. The table below shows the roles and their access profile defined in the application. A user can own several roles at a time.

Role name	Description
User	This role is automatically assigned to each user. It allows logon to MetaVoIP and use the its standard features
UserSiteAdmin	A user with the role UserSiteAdmin can manage the accounts of all users in his site.
UserAdmin	Users with the role UserAdmin can manage all user accounts irrespectively of the site they belong to.
SiteAdmin	This role provides the ability to manage data source drivers and PBX drivers of the own site. The "own site" is the site the user account belongs to.
Admin	Admins are able to create, change and delete site definitions, as well as add, change and remove data source drivers and PBX drivers for all sites.

The Application logic checks the authorization for each action a user triggers. If possible, the user GUI hides those actions, which are not allowed for a user.

3.5 MetaVoIP Data Model

MetaVoIP stores a replication of all data read-in from the sites' data sources in a local database. This has several advantages but also disadvantages. On the one hand MetaVoIP can access local data much quicker than data stored remotely in the organizations. This shortens the time necessary to display a new screen drastically. Furthermore data replication can be performed in off-peak hours using resources when they are not needed otherwise. On the other hand data replication always bears the thread of inconsistent and outdated data. As the remote data sources do not cooperate in the replication process (e.g., by writing a transaction log) the replication algorithms get more complex. Nevertheless the replication approach was chosen, as it is still less complex and the time to display data would be too long otherwise. Then setting up MetaVoIP a compromise has to be made between too frequent updates taking too many resources and too few updates leading to too much outdated data.

The data model of MetaVoIP in UML is depicted in Figure 5. The class *User* stores the attributes of a user. *Firstname*, *lastname*, *email* and *comments* are text fields storing the respective information. The field *userid* stores a site-unique identifier for each user. *Secret* stores a password, which is used for authentication in case the data source can read the password from the data source. See section 4.4 for a more detailed description on how this field is used. The class *Role* stores the roles of the users. Because a user might own more then one role an n:m-relationship has to be modeled. The class *PhoneAccount* represents one possibility to call a user, i.e. an extension. A user might be reachable by different means, e.g., by public switched telephone network (PSTN), Session Initiation Protocol (SIP) or mobile phone. The field *protocol* stores a description, which of these output channels a PBX should use when calling the user. *Accountid* stores the address information, which is protocol dependent, e.g., a phone number for PSTN or a SIP-URL for SIP.

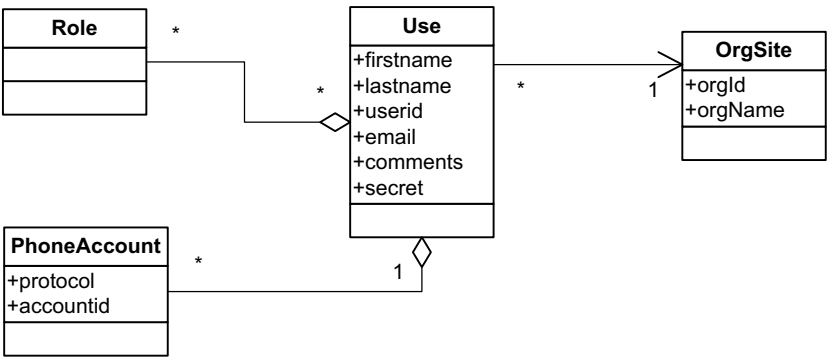


Figure 5. MetaVoIP data model

Each user is member of one site (class *OrgSite*). A site is identified by a unique identifier (*orgid*). The attribute *orgName* is only descriptive. The orgid in combination with userid from the class User is used to identify a user unambiguously.

4 Implementation

MetaVoIP is implemented as a web application using the programming language Java and an Apache Tomcat Server. Additionally, technologies from the Java Enterprise Edition are facilitated. Most prominent Java Server Pages (JSP) in combination with the Java Server Pages Standard Tag Library (JSTL) should be mentioned. The data model shown in section 3.5 is implemented using Java Beans. A bean each for User, OrgSite and PhoneAccount is defined. The roles are represented by a Java enumeration.

4.1 User GUI

The User GUI consists of web pages a user can access with his browser. This has the advantage that no additional software needs to be installed on client computers. The architecture described above is refined to implement the Model-View-Controller (MVC) concept using the framework Struts in version 1.3.8. A screen shot of the user GUI is shown in Figure 6. All screens are structured equally and they are divided into 5 parts: Part 1 displays a logo, part 2 offers links to log-in or log-off respectively. Part 3 is the navigation bar offering the user an easy way to navigate through the different functionalities of MetaVoIP. The navigation bar shows only those menu items the logged-in user is authorized for. Part 4 is the main part, displaying the actual content, e.g., the phone book. The information displayed depends on the menu item the user has chosen. The footer (part 5) displays status information. The User GUI features native language support. All text strings are separated from the source code and stored in a separate text file. Up to now translations to English and German already exist.

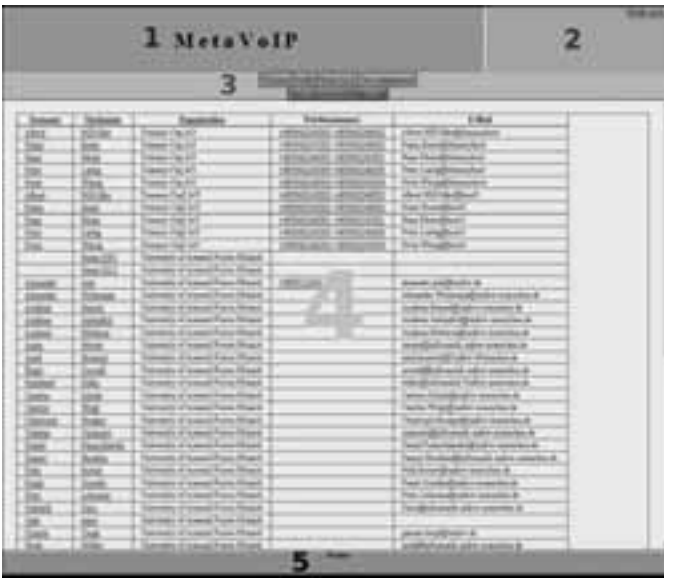


Figure 6. User GUI

4.2 Implementation of the Data Source Driver

The data source abstraction layer is realized by a class called MetaVoipNet and a Java interface that all data source drivers need to implement. MetaVoipNet is responsible for choosing the appropriate driver for a site, loading it and supervising its operation. The abstraction layer therefore does not only abstract from the type of data source, it also abstracts from the number of data sources and drivers. Up to now two data source drivers are implemented: A driver to read the user data from an LDAP server and a driver to read the data from Asterisk configuration files. The LDAP driver stores roles with objects of type “OrganizationalRole”. The names of the objects have to be the name of the role with the prefix “metavoip”. The object for the role “User” is therefore “cn=metavoipUser”. All owners of the role have to be listed in the attribute “roleOccupant”. Phone accounts are stored in LDAP using the LDAP schema of Astirectory [Ad08]. Astirectory is an extension to the Asterisk PBX of previous versions, which allows storing SIP and IAX accounts in an LDAP server instead of the respective configuration files.

The second driver retrieves its data directly from Asterisk configuration files. The driver periodically downloads the Users, SIP and IAX configuration files from an Asterisk server. The download is performed via Secure Copy (SCP) requiring an operational SSH daemon on the Asterisk server. MetaVoIP then parses these files and extracts the relevant data. The roles a user occupies are stored in the users.conf file as comments of a defined format.

4.3 Implementation of the PBX Driver

The PBX abstraction layer is realized by an abstract Java class, which already implements some common functionality. Drivers for a concrete PBX need to override this class and implement the abstract methods for establishing a call between two named phones and for getting the status of a phone account. Up to now a PBX driver for the Asterisk PBX is implemented. This driver uses the Asterisk Manager API for sending commands to the Asterisk PBX. The implementation uses the Asterisk-Java framework [Aj08], a Java wrapper library for the text based Asterisk Manager API. The Asterisk Manager API is quite unsecure because it operates only with clear-text passwords. Therefore a special proxy on the Asterisk server was implemented. Commands are transferred in a SSL-secured tunnel between the PBX driver on the MetaVoIP server and this proxy. The proxy decrypts the commands and forwards them to the Asterisk server.

4.4 Security

The security requirements to MetaVoIP are moderate. MetaVoIP does not implement business critical processes and the organizations' PBXs can run without MetaVoIP. However, MetaVoIP stores user passwords and passwords to telecommunication equipment, which need to be protected. Furthermore unauthorized access to MetaVoIP could be misused. Two possible ways exist to authenticate users. If a data source stores clear text passwords, which could be retrieved by the data source driver, the passwords will be replicated to the MetaVoIP server. In this case the data source driver has to ensure that the transfer is protected, e.g., the data source driver for the Asterisk configuration files transfers them with Secure Copy (SCP).

In the other case if no clear text passwords could be retrieved by the data source driver (because of technical or policy reasons), the driver checks the password against the data source each time a user logs in. The LDAP driver uses this method. When a user tries to log on to MetaVoIP, the driver checks the password by authenticating against the LDAP server of the user's site. If this authentication succeeds, name and password is correct and the user is granted access to the application. The Asterisk PBX driver implements an additional proxy, which is installed on the asterisk server in order to protect the connection. See Section 3 above for a description of this proxy.

5 Conclusions

MetaVoIP is a centralized server application, which can improve the integration between cooperating organizations. Its main focus is to loosely couple the PBX systems of the different organizations in order to make calls between organizations easier.

For this purpose, MetaVoIP uses a metadirectory approach that ensures short response time of the application. To avoid outdated information, data is automatically retrieved from data sources at the respective sites of the organizations. The organization keeps full control over its data and access rights.

MetaVoIP has been implemented as a prototype with two data source drivers (LDAP and Asterisk configuration files) and one PBX driver for Asterisk. The driver concept used for connecting PBX and data sources offers an easy way to implementing further drivers. Common functionality necessary for each PBX driver is already implemented in the base class. In addition, click-to-call functionality on a web site is also provided.

Next steps will be the implementation of additional drivers for data sources and PBXs. Also the migration of the LDAP data source driver to Asterisk release 1.6, which already includes LDAP support, will be a future task.

Acknowledgments

This research activity has been performed partially in the framework of the EU IST Network of Excellence EMANICS "Management of Internet Technologies and Complex Services" (IST-NoE-026854).

References

[Ad08] Asterisk e.V.: Astirectory. Available: http://www.asterisk-ev.org/projekte_astirectory.html

[AG08] voip-info.org: Asterisk GUI. Available: http://www.voip-info.org/wiki-Asterisk+GUI

[Aj08] Asterisk-Java. Available: http://asterisk-java.org/

[As08] Asterisk Flash Operator Panel. Available: http://www.asternic.org

[DS08] DS4J. Available: http://sourceforge.net/projects/ds4j/

[Ev08] Evidian Identity and Access Management. Available: http://www.evidian.com/iam/index.htm

[Fr00] Freire, J.; Lieuwen, D.; Ordille, J.; Garg, L.; Holder, M.; Urroz, H.; Michael, G.; Orbach, J.; Tucker, L.; Ye, Q.; Arlein, R.: MetaComm: a Meta-Directory for Telecommunications. In: Proc. 16th International Conference on Data Engineering, San Diego, California, USA, 2000, pp. 211-219.

[JE07] Jede, A.: Design and Implementation of a framework to integrate corporate Asterisk systems. Master thesis, Information Systems Laboratory, University of Federal Armed Forces Munich, 2007.

[LD97] Wahl, M.; Howes, T.; Kille, S.: Lightweight Directory Access Protocol (v3). Internet Engineering Task Force (IETF) RFC 2251, December 1997.

[Ma03] Masarati, P.: OpenLDAP & Meta-directory. Presented at the OpenLDAP Developers' Day, Vienna, Austria, July 2003. Available: http://www.openldap.org/conf/odd-wien-2003/ando.pdf

[Mi08] Microsoft Identity Lifecycle Manager 2007. Available: http://www.microsoft.com/windowsserver/ilm2007/default.mspx

[Or08] Oracle Virtual Directory. Available: http://www.oracle.com/technology/products/id_mgmt/ovds/index.html

[Pe08] Penrose Project. Available: http://penrose.safehaus.org

[Ra08] RadiantOne Virtual Diectory Server. Available: http://www.radiantlogic.com/

[Va05] VanMeggelen, J.; Smith, J.; Madsen, L.: Asterisk. The Future of Telephony. O'Reilly, 2005.

[Vo08] voip-info.org: LDAP. Available: http://www.voip-info.org/wiki/view/LDAP

Grid-Anwendungen

Ein zuverlässiger und schneller Dateitransfer mit dynamischer Firewall-Konfiguration für Grid-Systeme

Egon Grünter, Markus Meier, Ralph Niederberger, Thomas Oistrez

{e.gruenter,m.meier,r.niederberger,t.oistrez}@fz-juelich.de

Abstract: Firewalls separieren Bereiche mit verschiedenen Sicherheitsanforderungen. Diese Hauptaufgabe führt zu Problemen in Bezug auf Erreichbarkeit und Geschwindigkeit bei diversen Anwendungen. Besonders bei verteilten Systemen, wie z.B. einem Grid, ist eine ungehinderte Kommunikation, welche für die Nutzung verteilter Ressourcen unerlässlich ist, nicht möglich. Zudem nutzen Grid-Anwendungen oft mehrere und dynamisch reservierte Ports parallel. Dies führt zu der Aufgabe, Firewalls dynamisch zu konfigurieren. Dieser Artikel zeigt eine auf UDP hole punching basierende Lösung und beschreibt die Implementierung eines UNICORE Service, welcher diese Technologie nutzt um schnelle, direkte Dateitransfers durchzuführen.

1 Firewalls und Filterung von Datenverkehr

Firewalls werden genutzt um Bereiche mit verschiedenen Sicherheitsanforderungen von einander zu trennen. Die Hauptaufgabe besteht im Schutz der Rechner-Ressourcen vor unerlaubtem Zugriff und Missbrauch. Zur Erfüllung dieser Aufgabe verarbeitet die Firewall verschiedene Informationen, um zu entscheiden, ob eine Nachricht die Firewall passieren darf oder nicht. Der Firewall-Administrator definiert ein Regelwerk, welches die Implementierung der jeweiligen Sicherheitsrichtlinie darstellt. Der Netzwerkverkehr wird in Klassen von Paketen aufgeteilt, die weitergeleitet werden, und solche, die zurückgewiesen werden. Dieses Regelwerk dient als Basis für verschiedene Tests, die jedes eingehende Paket durchläuft. Die Firewall überprüft IP Addressen und Port Nummern des jeweiligen Protokoll-Headers. Sogenannte Statefull Inspection Engines nutzen außerdem den Verbindungsstatus.

Firewalls speichern Statusinformation haupsächlich für TCP-Ströme da TCP ein verbindungsorientiertes Protkoll darstellt. Verbindungen könne vier verschiedene Zustände haben: halb offen, offen, halb geschlossen, geschlossen. Es macht Sinn, zwischen den aktuellen Zuständen einer Verbindung zu unterscheiden. Zudem ist TCP ein zuverlässiges Protokoll. Das bedeutet, dass ein Paket, welches durch fehlende Bestätigungen vom Empfänger als verlohren erkannt wurde, nochmals gesendet wird. Weitere Informationen hierzu finden sich in [Ste94].

Das User Datagramm Protocol (UDP) ist ein Transportprotokoll und ist weder zuverlässig noch verbindungsorientiert. Eine Anwendung, die UDP nutzt, muss sicherstellen, dass die Daten erfolgreich übertragen wurden. Obwohl bei UDP keine Verbindungen existieren,

nutzen Firewalls einen einfachen Mechanismus, um Verbindungen zu simulieren. Abbildung 1 zeigt einen Client hinter einer Firewall, der UDP Pakete nach außen senden darf und ein UDP Datagram an einen DNS Server außerhalb des Firmennetzwerkes sendet.

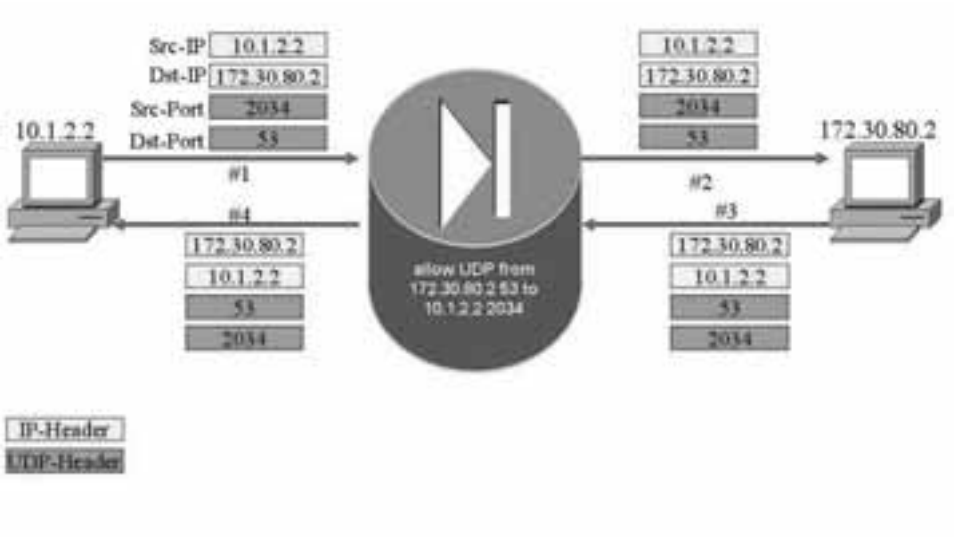


Abbildung 1: Firewalls und UDP

Der Client generiert das UDP Datagram und sendet es an die Firewall (#1). Die Firewall kontrolliert das Datagram und leitet es an das Ziel weiter (#2). Anhand der gesammelten Informationen fügt die Firewall einen Eintrag zur Verbindungstabelle hinzu, der die IP-Addressen und Port-Nummern von Quelle und Ziel enthält. Dies führt zu der folgenden, dynamisch generierten Zugangsregel, welche Antwortpakete des Servers an den Client erlaubt:

```
allow UDP from 127.30.80.2 port 53 to 10.1.2.2 port 2034
```

Diese ist für eine bestimmte, konfigurierbare Zeit gültig. Die Regel stellt sicher, dass innerhalb der festgelegten Zeit die Antworten vom Server (#3) die Firewall passieren können (#4).

2 Grid Anwendungen und Firewalls

Ein Grid ist ein verteiltes System, welches seinen Nutzern Zugriff auf Rechenzeit, Speicherplatz und verteilt vorliegende Daten bietet. Es bildet einen Bund aus verschiedenen, geographisch verteilten unabhängigen Organisationen, auch als virtuelle Organisation bezeichnet. Auf die verfügbaren Ressourcen kann statisch oder dynamisch zugegriffen werden, je nach Anforderung der Nutzer bzw. der Anwendung.

Viele Grid-Anwendungen benötigen hohe Bandbreiten und kurze Verzögerungszeiten. Zudem werden große Datenmengen übertragen, die das Ziel schnell und zuverlässig erreichen müssen. Ein Ansatz dazu ist die Nutzung mehrerer paralleler TCP-Verbindungen. GridFTP [web06] ist ein Beispiel hierfür: Es setzt voraus, dass mehrere Verbindungen zwischen Sender und Empfänger hergestellt werden. Das Protokoll sieht vor, dass ein Server, der GridFTP nutzt, auf einer Vielzahl von Ports von außen erreichbar ist. Das statische und somit langfristige Öffnen großer Portbereiche (in der Regel 5000 Ports) an der Firewall führt zu ernsten Sicherheitsproblemen, da die Ports von bösartiger Software genutzt werden können, während GridFTP sie nicht verwendet. GridFTP ist ein schnelles Verfahren, aber durch die statische Port-Öffnung an der Firewall ist seine Verwendung in den meisten produktiven Umgebungen nicht gerne gesehen.

Zur Erhöhung der Sicherheit scheint eine dynamische Konfiguration von Firewalls Sinnvoll. Diese sollte folgende Anforderungen erfüllen:

1. Sie kann reibungslos in die existierende Sicherheitsrichtlinien integriert werden.
2. Sie kann in Open-Source und kommerziellen Produkten genutzt werden.
3. Sie erlaubt die Kommunikation zwischen den beiden Partnern nur für die minimal nötige Dauer.

Zur Zeit gibt es verschiedene Lösungen um eine Firewall dynamisch zu konfigurieren. Diese sind entweder proprietär und herstellerspezifisch wie in den Cisco PIX Produkten [Cis], oder sie unterstützen nur bestimmte Arten von Firewalls. CODO [SAL05] ist eine solche Lösung. Neue universielle Ansätze zur Konfiguration und neue Protokolle zur Signalisierung befinden sich in der Entwicklung, sind aber noch nicht fertiggestellt bzw. noch unzureichend verbreitet. Dieser Artikel stellt einen neuen Ansatz zur dynamischen Firewall-Konfiguration vor. Er nutzt den Mechanismus des UDP hole punching, einer üblichen Technik im NAT [Sch06] Bereich. Der Ansatz ist leicht zu implementieren und mit allen zur Zeit üblichen Firewalls sofort einsetzbar. Da das Verfahren nur mit UDP funktioniert, und somit in Kombination mit etablierten Protokollen wie GridFTP nicht funktioniert, ist auch die Wahl eines geeigneten Transfer-Protokolls nötig.

3 UDP hole punching

UDP hole punching ist eine Methode zur Herstellung bidirektionaler UDP Verbindungen zwischen zwei durch Firewalls getrennte Internet Hosts. Das Verfahren basiert auf der Simulation von UDP Verbindungen durch Firewalls. Eingesetz wird UDP hole punching bereits seit längerem in VoIP- und P2P-Software. Hier dient es der Herstellung von direkten Verbindungen zwischen Endpunkten, die sich oft hinter NAT-Routern befinden. NAT-Router sind nur schwierig dynamisch zu konfigurieren, da sie die Portnummern und IP-Adressen der weitergereichten Pakete ändern und die Endsysteme somit ihre eigenen Verbindungsdaten nicht kennen. Im Grid-Umfeld spielt NAT keine besondere Rolle, so dass das Verfahren hier zu der im folgenden beschriebenen Methode vereinfacht werden

kann, was einerseits der Robustheit und der Sicherheit zu Gute kommt, andererseits einen Einsatz mit NAT-Routern aber unmöglich macht.

Es gelten folgende Voraussetzungen zur Nutzung von UDP hole punching:

- Die lokale Firewall erlaubt ausgehende UDP Verbindungen
- Die lokale Firewall simuliert diese UDP Verbindungen wie in Kapitel 1 beschrieben
- Es existiert ein zentraler Relay Server

Der zentrale Server ist ein wesentlicher Bestandteil des Konzeptes. Jeder Client verbindet sich zu diesem Server mit einer dauerhaften TCP Verbindung. Im Folgenden wird beispielhaft beschrieben, wie zwei Hosts aus verschiedenen Sicherheitsbereichen eine Verbindung mittels UDP hole punching herstellen (Abbildung 3). Der Initiator (Client A) sendet eine Nachricht an den Relay Server (Nachricht #1). Diese enthält die Information, das Client A mit Client B kommunizieren will und dafür einen bestimmten UDP Port, z.B. 4711, nutzen wird. Der Server benachrichtigt Client B über den Verbindungswunsch und übersendet die IP Addresse und den UDP Port von Client A (#2). Client B sendet seinen eigenen UDP Port, z.B. 8822 an den Server und sendet gleichzeitig ein erstes UDP Paket von Port 8822 an den Port 4711 des Client A (#3).

Die lokale Firewall von Client B leitet das Paket weiter und erstellt einen Eintrag in der Verbindungsdatenbank sowie die dynamische Zugangsregel, die eine Antwort von Client A zulässt. Die Firewall von Client A verwirft das Paket, da es sich hier um eine von außen initiierte Verbindung handelt. Der Server informiert Client A über die TCP Verbindung, dass Client B an Port 8822 erreichbar ist (#4).

Client A sendet jetzt ein UDP Paket von Port 4711 an B's Port 8822 (#5). A's lokale Firewall erstellt einen entsprechenden Verbindungseintrag. A's Paket wird von B's Firewall an B weitergeleitet, da die ursprünglich von B ausgegangene Verbindung in B's Firewall noch gültig ist. Jetzt ist eine Verbindung zwischen A und B aufgebaut, obwohl die Regelwerke beider Firewalls normalerweise keine eingehenden Verbindungen zulassen würden.

Bei diesem Verfahren ist nur UDP basierter Datenaustausch möglich. Da bei der Nutzung von TCP die Sequenznummern von der Firewall mit überwacht werden, und diese sich bei unterschiedlichen Verbindungen unterscheiden, kann TCP als Transportmechanismus nicht genutzt werden.

4 Ein Dateitransfer auf Basis von UDP hole punching

Für einen Dateitransfer, der auf UDP hole punching basiert, stellt die Grid-Software UNICORE [UNI06] eine ideale Umgebung dar. Viele der im letzten Kapitel beschriebenen Konzepte sind bereits realisiert. Es gibt bereits ein Gateway für alle Kontrollnachrichten, die mit X.509 Zetifikaten verschlüsselt werden. Daher passt sich das Verfahren sehr gut in die Archtitektur ein. Ein dedizierter Vermittlungsserver für das UDP hole punching ist

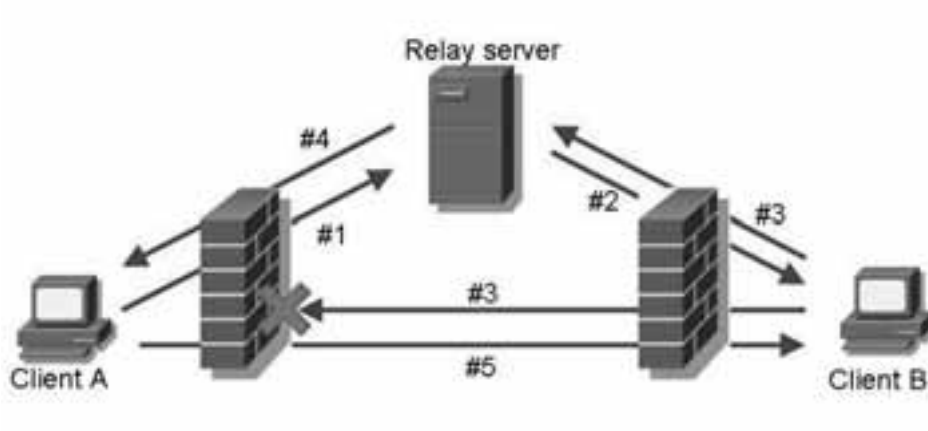


Abbildung 2: UDP hole punching

durch die Nutzung des UNICORE Gateways nicht mehr nötig. Benutzer und Geräte werden zentral autorisiert und authentifiziert. Die existierenden Dateitransfers in UNICORE basieren alle auf vordefinierten Basis-Klassen die grundlegende Funktionen und Fehlerbehandlung implementieren. Eine neue Transfermethode wird von diesen Klassen abgeleitet und erweitert sie um die für das Verfahren spezifischen Methoden. Daher besteht die Implementierung des Transfers aus der Bereitstellung der Methoden zum Austausch der Verbindungsdaten wie IP-Addresse und UDP-Port über das UNICORE Gateway, zum Versand des hole-punching Paketes und zum Datenaustausch über UDP-Pakete.

Eine neue Aufgabe ergibt sich aus der Verschlüsselung von UDP-Verkehr. Eine Möglichkeit ist die Verschlüsselung durch einen einfachen symmetrischen Algorithmus. Da die TCP-Kontrollverbindung zwischen Client und Server über das Gateway mit X.509 Zertifikaten verschlüsselt ist, kann der gemeinsame Schlüssel darüber ausgetauscht werden. Diese Verschlüsselung des Datentransfers sollte optional sein, um bei unsensiblen Daten eine höhere Geschwindigkeit bei gleichzeitig geringerer CPU-Belastung errreichen zu können.

Eine weitere Aufgabe ergibt sich aus den Anforderungen der Grid-Anwendung an den Dateitransfer. Dieser sollte schnell und zuverlässig sein. Da bei reiner UDP Kommunikation keine zuverlässige Übertragung garantiert ist, muss die Anwendung sicherstellen, dass alle Pakete in der richtigen Reihenfolge ankommen. Die Grid-Middleware braucht also ein Protokoll auf Anwendungsebene, das Zuverlässigkeit implementiert.

Dies kann vom `UDP-based Data Transfer Protokoll` (UDT) übernommen werden. UDT benutzt UDP als Transportprotokoll und ist somit „UDP hole punching“ fähig, garantiert aber Zuverlässigkeit durch eine zusätzliche Protokollschicht. Das folgende Unterkapitel beschreibt die allgemeinen Eigenschaften von UDT [GG04].

4.1 UDP-based Data Transfer Protocol (UDT)

Entwickelt wurde UDT am National Center for Data Mining der University of Illinois at Chicago [GG04]. Ziel war es, ein Protokoll zu entwickeln, das besonders auf schnelle moderne Netze zugeschnitten ist und diese vollständig ausnutzen kann. UDT ist in C++ implementiert. Es bietet einen eigenen Namensraum und orientiert sich streng an der standard Socket-API. Es braucht daher keinen großen Aufwand zur Einarbeitung und kann mit nur wenigen Anpassungen in bestehende Projekte integriert werden. Die wesentlichen Eigenschaften von UDT sind:

Anwendungsebene: Da UDT als Bibliothek auf Anwendungsebene implementiert ist, kann es von jeder Anwendung benutzt werden. Es sind keine Veränderungen im Kernel bzw. im TCP/IP-Stack der Systeme notwendig. Die Anwendungen brauchen auch keine administrativen Rechte auf den Systemen.

Systemunabhängig: Die Bibliothek ist sowohl für Linux als auch für Windows verfügbar.

Open-Source: Die Bibliothek ist bis einschließlich Version 3.3 unter der „GNU Library General Public License“ (LGPL) verfügbar. Ab Version 4 wird sie unter der BSD Lizenz veröffentlicht. Damit ist eine ausreichende Nutzbarkeit sowohl für Open-Source-Projekte als auch für kommerzielle Produkte sichergestellt.

Die wichtigen Eigenschaften und Funktionen Verbindungsorientiertheit, Zuverlässigkeit, Stau- und Flusskontrolle implementiert UDT auf Anwendungsebene mit den gleichen Mechanismen wie TCP. Da die Staukontrolle bei UDT modular aufgebaut ist, kann man als Entwickler eigene Algorithmen implementieren. Diese werden über Callback-Interfaces in die UDT-Bibliothek eingebunden. UDT bringt bereits einen Algorithmus zur Staukontrolle mit. Er erreicht bereits nach kurzer Zeit die maximale Datenrate des genutzten Netzes und reagiert auf vereinzelte Paketverluste wesentlich weniger stark als TCP. Trotzdem verhalten sich mehrere UDT-Ströme in einem Netz fair zueinander. TCP-Ströme werden von diesem Verfahren zwar weniger stark beeinträchtigt als es bei unkontrolliertem UDP-Verkehr der Fall wäre, aber „TCP-friendly“ ist das Verfahren nicht. Im praktischen Einsatz empfiehlt sich daher die Verwendung einer alternativen Staukontrolle oder die Limitierung der Bandbreite durch Quality-of-Service Regeln in den Routern. Dies gilt besonders bei der Nutzung öffentlicher Netze, in denen TCP-freundliches Verhalten Pflicht ist. Oft werden im Grid-Umfeld aber auch dedizierte Netze für den Grid-Verkehr genutzt. Dort ist UDT mit der standard-Staukontrolle besonders gut geeignet.

5 Die Gesamtarchitektur

Nachdem das UDP hole punching und seine Verwendung im Grid-Umfeld im letzten Kapitel beschrieben wurden, kann nun die Implementierung des neuen Dateitransfers im Gesamtsystem genauer betrachtet werden.

Die Implementierung basiert auf den beiden UNICORE-Klassen „FileTransferClient“ und „FileTransfer“ die nur durch eine zusätzliche WebService-Methode „initUDT“ erweitert wird. Wenn ein Transfer gefordert wird, dann erstellt die UNICORE-Middleware die Transferobjekte (WebService-Ports) auf Client- und Server-Seite. Der Transfer-Prozess besteht dann aus den folgenden Schritten. Der Client bereitet zuerst die Verbindung vor, indem er seinen UDT-Socket fest an einen zufällig gewählten Port bindet. Dann ruft er die Methode „initUDT“ des Servers auf. Diese Aufrufe werden nicht direkt getätigt, sondern auf dem Umweg über das Gateway. Seine IP und Port-Nummer sendet der Client als Parameter der WebService-Methode. Der Server bereitet seinen eigenen UDT-Server-Socket vor und führt das UDP hole punching aus, indem er ein leeres UDP-Paket an die IP-Port-Kombination des Client sendet. Dann sendet er seine eigene IP und Port-Nummer als Rückgabewert der Methode an den Client zurück. Der Client kann nun mit seinem UDT-Socket eine Verbindung zum Server aufbauen. Über diese Verbindung wird der Inhalt der zu transportierenden Datei mittels einfacher Send-Recieve-Funktionen übertragen. Informationen wie Dateiname und Dateigröße werden unabhängig vom Transfer-Verfahren von UNICORE festgestellt und übermittelt.

Da UDT in C++ geschrieben ist, UNICORE aber in JAVA, ist eine hybride Lösung nötig. Ein erster Ansatz bestand aus der Verwendung des JAVA Native Interface (JNI) für den Aufruf der UDT-Methoden. Ein flexibleres Verfahren stellt aber die Auslagerung des Transfers in eine eigene, in C++ geschriebene Anwendung dar. Diese wird von den Transfer-Objekten in einem zusätzlichen Prozess gestartet. Die Übergabe der Verbindungsparameter und zusätzlicher Informationen erfolgt durch die Umleitung und Manipulation der Standardein- und -ausgabe durch das JAVA-Objekt. Das UDP hole punching wird ebenfalls in dieser Anwendung durchgeführt. Die Anwendung muss bei diesem Vorgehen für das entsprechende System kompiliert sein. Sie ist nicht System- und Hardwareunabhängig. Durch die Trennung kann das Verfahren noch leichter in Kombination mit anderer Middleware genutzt, oder so sogar von Hand bedient werden.

Der Dateitransfer kann als Alternative zu den bisher üblichen Dateitransfermethoden in UNICORE genutzt werden. In produktiven Netzen zeigt er in Bezug auf Netzwerksicherheit und Geschwindigkeit die oben beschriebenen Vorteile gegenüber anderen Verfahren.

Für den Einsatz von GridFTP ist, verglichen mit dem neuen Verfahren, eine weniger sichere Konfiguration nötig, wärend die UNICORE Verfahren ByteIO und BFT die Dateien nicht direkt zwischen den Systemen austauschen, sondern durch WebService-Calls bzw. zusätzliche HTTP-Verbindungen über das Gateway. Sie sind wesentlich langsamer als direkter Austausch, da das Gateway in der Regel einen Bottleneck bildet.

6 Zusammenfassung

Firewalls sind zur Sicherung eines Netzwerkes absolut erforderlich. Der Einsatz von Sicherheitsregeln auf den Firewalls führt aber zu einer Einschränkung der Konnektivität. Grid-Anwendungen sind hier von besonders betroffen. Sie benötigen hohen Durchsatz und geringe Latenzen. Die Nutzung paralleler Verbindungen ist in diesem Zusammenhang

üblich. Dazu werden große Port-Bereiche an den Firewalls statisch geöffnet, was eventuell zu Sicherheitsproblemen führen kann. Dynamische Konfiguration kann dieses Problem mildern.

In diesem Bericht wurde eine Lösung vorgestellt, die eine Firewall mittels UDP hole punching auf sichere Weise dynamisch konfigurieren kann. Da dies nur mit UDP möglich ist wurde UDT als zuverlässiges, UDP basiertes Transport-Protokoll genutzt. Das Konzept konnte für die Verwendung im Grid Bereich angepasst und vereinfacht werden. Eine praxistaugliche Implementierung für UNICORE wurde vorgestellt. Diese Lösung bietet ein in vielen Organisationen ausreichendes Sicherheitsniveau.

Zu beachten ist, dass der standardmäßig mitgelieferte Algorithmus zur Staukontrolle für öffentliche Netze zu aggressiv vorgeht und die anderen Verbindungen im Netz stark unterdrückt. Ein faireres Verfahren sollte je nach Einsatzzweck genutzt werden. Alternativ kann die verwendete Bandbreite durch Quality of Service Regeln in den Routern der beteiligten Netze begrenzt werden.

Das Verfahren ist durch die modulare Implementierung leicht in anderen Grid-Anwendungen einsetzbar und funktioniert mit allen üblichen Firewalls. Es kann als Übergangslösung dienen, bis „echte“ dynamisch konfigurierbare Firewalls verfügbar sind bzw. neuartige Protkolle mit entsprechenden Features standardisiert und verbreitet wurden.

Literatur

[Cis] Cisco. *Cisco Security Appliance Command Line Configuration Guide - For the Cisco ASA 5500 Series and Cisco PIX 500 Series Software Version 7.2*.

[GG04] Y. Gu and R.L. Grossmann. *UDT: A transport protocol for data intesive applications*. University of Illinois at Chicago, August 2004.

[SAL05] S. Son, B. Allcok, and M. Livny. CODO: Firewall Traversal by Cooperative On-Demand Opening. In *14th IEEE Symposium on High Performance Distributed Computing (HPDC14)*, http://www.cs.wisc.edu/ sschang/papers/CODO-hpdc.pdf, July 2005.

[Sch06] J. Schmidt. Der Lochtrick - Wie Skype & Co. Firewalls umgehen. *Ct*, 17:142 ff, 2006.

[Ste94] W. Richard Stevens. *TCP/IP Illustrated*. Addison Wesley, 1994.

[UNI06] UNICORE. UNICORE Grid computing Technology, August 2006. http://www.unicore.eu/.

[web06] Globus Toolkit website. *http://www.globus.org/toolkit/docs/4.0/data/gridftp*, August 2006.

Nachhaltigkeitsstrategien bei der Entwicklung eines Lernportals im D-Grid

Viktor Achter, Marc Seifert, Ulrich Lang,
Joachim Götze, Bernd Reuther, Paul Müller

{vachter, marc.seifert, lang}@uni-koeln.de,
{j_goetze, reuther, pmueller}@informatik.uni-kl.de

Abstract: In dem Projekt SuGI (Sustainable Grid Infrastructures, gefördert durch das BMBF) wird unter anderem ein Lernportal entwickelt, welches auf die besonderen Anforderungen des D-Grids ausgerichtet ist. In diesem Umfeld spielen die heterogenen Communities, sowie die Nachhaltigkeit der Prozesse und der erstellten Produkte eine zentrale Rolle. Darunter fallen unter anderem die zum Teil sehr unterschiedlichen Vorkenntnisse von Anwendern und Grid-Experten, sowie die ressourcenschonende Bereitstellung und Archivierung von Inhalten und Erkenntnissen aus dem Gridumfeld oder die unterschiedlichen Zielsetzungen zwischen Rechenzentren von KMUs und solchen von etablierten Forschungsinstitutionen.
Dieser Beitrag beschreibt die Konzeption und das Vorgehen bei der Entstehung des D-Grid Lernportals. Hierbei wurde der Entwicklungsprozess auf die speziellen Anforderungen von Lernportalen ausgerichtet, welche vor allem in der Dynamik der Inhalte und den komplexen Bedürfnissen der heterogenen Grid-Communities sowie den vielfältigen Formen des Lernens liegen. Dazu wurde ein evolutionäres Vorgehensmodell gewählt, wobei mehrere Generationen in einem durch Feedback unterstützten, rekursiven wie auch iterativen Prozess entstehen. Abschließend werden Ergebnisse der aktuellen Evaluierung des Lernportals präsentiert.

1 Einleitung

SuGI (Sustainable Grid Infrastructures) ist ein Projekt des D-Grid [DGR], der deutschen Grid Initiative und wird durch das BMBF (Bundesministerium für Wissenschaft und Forschung) gefördert. Die Kernaufgabe von SuGI besteht darin, Grid bzw. Wissen über Grid-Technologien breitenwirksam verfügbar und nutzbar zu machen. SuGI ist somit auf eine Vielzahl an Rechenzentren von Hochschulen und Unternehmen ausgerichtet, die Grid-Technologien bisher nur in geringem Maße oder gar nicht nutzen. Im Verlauf des Projekts werden die im D-Grid erlangten Erkenntnisse sowie D-Grid-relevante Inhalte in geeigneter Weise den verschiedenen Zielgruppen[1] zugänglich gemacht, wie im Folgenden näher erläutert wird. Dazu bietet SuGI einen Katalog an Maßnahmen. Neben Präsenzschulungen und der Bearbeitung rechtlicher Aspekte rund um den Einsatz von Grid und Grid-Technologien finden sich darunter auch die Bereitstellung von Konfigurationswerkzeugen und Übungs- bzw. Produktivsystemen für die im D-Grid unterstützten Grid-Middlewares. Eine

[1] Vgl. http://portal.sugi.uni-koeln.de/de/ueber-sugi/zielgruppen.html

weitere, wesentliche Maßnahme besteht im Aufbau eines Lernportals[2], über das Schulungen, Lernmodule, Übungssysteme etc. sowie Videoaufzeichnungen von Präsenzveranstaltungen online abrufbar sind. Der Fokus liegt dabei auf gut skalierenden Methoden des E-Learning, um mit hoch qualitativen Schulungsmaterialien vor allem Multiplikatoren an Rechenzentren die Möglichkeit zur Aus- und Weiterbildung von Mitarbeitern und Anwendern zu geben. In diesem Umfeld spielen die heterogenen Communities, sowie die Nachhaltigkeit der Prozesse und der erstellten Produkte eine zentrale Rolle. Darunter fallen unter anderem die zum Teil sehr unterschiedlichen Vorkenntnisse von Anwendern und Grid-Experten, sowie die ressourcenschonende Bereitstellung und Archivierung von Inhalten und Erkenntnissen aus dem Grid-Umfeld oder die unterschiedlichen Zielsetzungen zwischen Rechenzentren von Unternehmen oder Forschungsinstitutionen. Dieser Beitrag beschreibt die Konzeption und das Vorgehen bei der Entstehung des SuGI-Portals. Der Entwicklungsprozess wurde dabei speziell auf die Anforderungen von Lernportalen ausgerichtet, die im Wesentlichen in der Dynamik der Inhalte und den komplexen Bedürfnissen der heterogenen Grid-Communities sowie den vielfältigen Formen des Lernens liegen (vgl. [GW07]). Dazu wurde ein evolutionäres Vorgehensmodell gewählt, wobei mehrere Generationen in einem durch Feedback unterstützten, rekursiven wie auch iterativen Prozess entstehen. Dieser Ansatz ist insofern innovativ, da er bewährte Modelle der Softwareentwicklung mit rezenten Konzepten der Entwicklung von E-Learning-Portalen verknüpft [GW07]. Das Ergebnis ist ein Lernportal, das an der Schnittstelle zwischen organisierter Darstellung von Information und methodenbasiertem E-Learning steht. Vor allem die Unterstützung einer Vielzahl von Inhalten unterschiedlicher Formate bzw. für verschiedene Zielgruppen, das Qualitätsmanagement und der Einfluss von Evaluierungsergebnissen in den weiteren Entwicklungsprozess tragen zu einem robusten Nachhaltigkeitskonzept bei. Der Erfolg des SuGI-Portals mit derzeit über 190 Schulungsinhalten und durchschnittlich mehr als 3000 Seitenaufrufen pro Monat sowie positiven Evaluierungsergebnissen bestätigt diesen Ansatz. Exemplarisch wird dazu die Evaluierung des SuGI-Portals, sowie deren Ergebnis ausführlich beschrieben.

2 Evolutionäres Vorgehen sowie Qualitätsmanagement als Nachhaltigkeitsstrategie bei der Entwicklung eines Lernportals

Softwareentwicklungsprozesse stellen seit je her eine besondere Herausforderung dar. Dies wird beispielsweise belegt durch den regelmässig von der Standish Group durchgeführten Chaos Report [Gro04], der seit 1994 in einem zweijährigen Rythmus erscheint. Demnach erreichen etwa ein Viertel aller IT-Projekte nicht ihre Ziele. Im Gegensatz zu vielen Projekten aus anderen Bereichen hat man es hier in aller Regel mit Innovationsprojekten [Bal98] zu tun, die sich durch einen besonders hohen Risikograd auszeichnen. Dies wird nicht zuletzt bedingt durch den geringen Informationsgrad über zukünftige Ereignisse. In der Regel liegen nur wenige Erfahrungen vor, aus denen erprobte Handlungsempfehlungen abgeleitet werden könnten. Die Entwicklung eines Lernportals steht dieser Problematik in keiner Hinsicht nach. Gegenteilig kommen hier erschwerend die mannigfaltigen

[2] http://sugi.d-grid.de

Abhängigkeiten von externen Einflüssen hinzu. Im Gegensatz zu internen Einflüssen, die durch die Projektorganisationsstruktur bedingt im institutionalisierten Einflussbereich der Projektkoordination stehen sind externe Einflüsse außerhalb des Selben und daher generell schwer zu kontrollieren. Sie stellen somit ein besonderes Risiko dar. Weiterhin definieren sich die Anforderungen nicht aus bekannte Arbeitsabläufe und einer überschaubare Menge von Nutzern, was eine stichhaltige Anforderungsanalyse erschwert.

Hieraus folgt, dass der Entwicklungsprozess eines Portals mit einem besonders angepassten und stetig kontrollierten Projektplan einher gehen muss. So steht am Anfang eine gründliche Analyse der externen sowie der internen Einflußgrößen. In diesem Kontext sind vor allem die Ressourcenrestriktionen, die geographisch verteilten Projektpartner und die damit verbundenen kommunikativen Restriktionen, als auch die heterogenen Zielgruppen zu nennen. Wie bereits angedeutet sind die Zielgruppe eines Portals üblicherweise mindestens eine große Community, die aus vielen unterschiedlichen Individuen mit unterschiedlichen Geschmäckern, Bildungsgraden und Erfahrungen besteht. Dies erschwert die Konzeptionierung nicht unerheblich, wie später deutlich werden wird.

Um dieser Problematik zu begegnen, eignen sich evolutionäre Entwicklungsmodelle in besonderem Maße. Diese verfolgen einen prototypischen Ansatz, der sich an die Vorgehensweisen des Changemanagement [DL00] anlehnt. Abbildung 1 als Auszug von [Bal98] beschreibt ein einfaches Modell, bestehend aus drei Schritten, welches als Grundlage für den iterativen Anpassungsprozess von innovativen Software Produkten dient. Es beschreibt einen durch Rückkopplung geregelten Zyklus, durch den das Produkt von einem stabilen Zustand (Generation) zu dem nächsten übergeleitet wird.

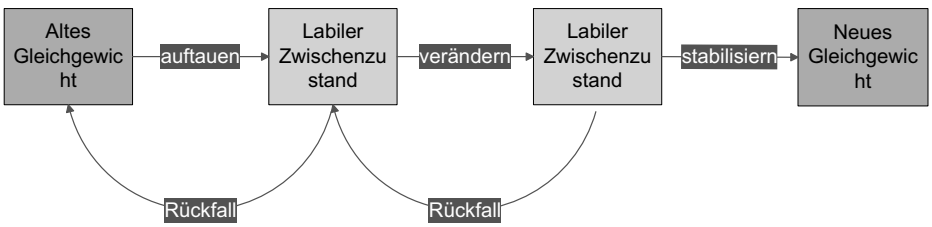


Abbildung 1: Drei-Schritte-Modell der Dynamik von Veränderungsprozessen

Die Darstellung in Abbildung 1 beschreibt auf generische Art und Weise die Prozedur, wie ein Produkt von einem stabilen Zustand in einen folgenden stabilen Zustand überführt werden kann und stellt eine wesentliche Grundlage für die modernen iterativen Entwicklungsmodelle dar. So wird in einem ersten Schritt durch Auftauaktivitäten der stabile Zustand in einen Entwicklungszustand überführt, der fortan auf einer dedizierten Entwicklungsinstanz betrieben wird. Dieser befindet sich in den Entwicklungsphasen in einem labilen Zustand. Nach diversen Iterationszyklen, in denen die Funktionalität verändert wird, folgt eine Stabilisierungsaktivität. In dieser Aktivität werden die funktionalen Aspekte nicht mehr erweitert, sondern nur noch konsolidiert. Das Resultat ist ein veränderter stabiler Zustand, welcher wieder auf der Produktivinstanz betrieben werden kann.

So wird in kurzen Abständen phasenweise ein Vorgehensmodelle angewendet, um das Produkt in Zwischenstadien (Prototypen) zu versetzen, die es der Nutzerschaft ermöglichen, ihre Vorstellung mit der Inkarnation des Portals, dem Prototypen, zu vergleichen und einen Eindruck davon zu erhalten, ob sich Ihre Vorstellungen von Attributen eines guten Produktes bewahrheiten. Somit gliedern sich die Arbeitsblöcke einer jeden Generation grob in die Schritte:

- Anforderungsanalyse
  Die Anforderungsanalyse setzt den Grundstein für die folgenden Entwicklungen. Hier konkretisieren sich die Anforderungen der Nutzer, um die definierten Ziele zu erreichen. Dieser Prozess gestaltet sich, wie eingangs erwähnt, bei der Portalentwicklung besonders Aufwändig, da eine große Anzahl von Nutzern mit unterschiedlichen Interessenslagen und Vorbildungen zu berücksichtigen sind. Im Rahmen des Projektes SuGI werden hierfür projektintern Fallstudien untersucht, sowohl offenen, als auch geschlossene Befragungen durchgeführt, sowie Nutzerfeedbacks eingeholt und ausgewertet.

- Design/Entwurf
  Im folgenden Schritt werden die unterschiedlichen Stadien der Konzeption durchlaufen. Hierbei wird im besonderen Maße darauf geachtet, dass Änderungen verträglich zu den bisherigen Entwicklungen integrierbar bleiben, wie auch die Integrierbarkeit in die externe D-Grid Struktur. In einem weiteren Schritt werden die Entwurfsvorgaben in kleine Arbeitspakete aufgeteilt und nach einer Aufwandsschätzung an projektinterne Entwicklergruppen verteilt.

- Implementierung
  Die Inplementierung folgt nach Möglichkeit in kleinen Gruppen (Organisationseinheiten), die untereinander mit geringem Aufwand kommunizieren können – üblicherweise Gruppen, die geographisch nah beieinander sind. Zur organisations-interne Kommunikation, wie auch die Abstimmung zwischen den Organisationseinheiten stehen Werkzeuge aus dem Bereich der Groupware (MS Sharepoint), sowie Versionierungswerkzeugen (SubVersion) zur Verfügung. Die Entwicklung erfolgt auf einem weitgehend entkoppelten Entwicklungsportal.

- Veröffentlichung
  Nach einer internen Qualitätssicherung, in der in Gruppen diskutiert wird, welche Ziele in welchem Maße erreicht wurden findet die Veröffentlichung der weiterentwickelten Generation des Portals statt.

- Rückkopplung
  Am Ende eines jeden Entwicklungszyklus finden Befragungsaktionen zu den Releases statt. Diese erfolgen sowohl gezielt auf Basis von Musternutzer, wie auch durch Feedbackmöglichkeiten im Portal selber und durch Fragebogenaktionen. Flankierend hierzu werden in Kooperation mit Vertretern anderer D-Grid Projekte im Rahmen von Workshops und anderen Veranstaltungen abschließende Gesprächs- und Diskussionsrunden durchgeführt (Dialog), bei denen das Nutzungserlebnis hinterfragt wird.

Durch dieses Vorgehen ist es möglich, in kürzeren und regelmäßigen Abständen prototypisch zu überprüfen, welche Aspekte den Anforderungen der Nutzerschaft Rechenschaft tragen, und welche dies nicht tun. So kann erreicht werden, dass Fehlentwicklungen innerhalb des Projektes schnell entdeckt und korrigiert werden können und so ressourcenschonend entwickelt wird.

Flankierend hierzu stellen Maßnahmen zur Qualitätssicherung und zum Qualitätsmanagement einen wesentlichen Aspekt zur Steigerung der Qualität und somit auch der Nachhaltigkeit von Lernportalen dar. Gemäß [BRNP04] setzt sich „zunehmend die Einsicht durch, dass die konsequente Umsetzung des Qualitätsmanagements die Qualität messbar steigert, Kosten langfristig senkt und über eine höhere Motivation der Mitarbeiter Innovationen anregt." Neben der Informationsverarbeitung, in der schon seit einigen Jahren Methoden des Qualitätsmanagements zum Einsatz kommen, wurde diese Notwendigkeit in jüngerer Vergangenheit zunehmend auch für das Bildungswesen bzw. den Bereich der formellen und informellen Aus- und Weiterbildung erkannt. Dies belegen unter anderem die Veränderungsprozesse in Schulen und Universitäten (z.B. die Evaluierung, Akkreditierung und Zertifizierung von Studiengängen, Fachbereichen und extracurricularen Aus- und Weiterbildungsprogrammen) in den letzten Jahren. Für das von SuGI entwickelte Lernportal im Bereich E-Learning zu Grid-Technologien bedeutet dies neben der strategischen und konzeptionellen Planung sowie der Implementierung auch die Durchführung von Evaluierungsmaßnahmen, auf deren Basis die Wünsche und Zufriedenheit der Zielgruppen festgestellt und darauf aufbauend eine kontinuierliche Verbesserung der angebotenen Dienstleistungen gewährleistet werden kann (vgl. [Wal95, Hof95, Dör03, BRNP04, BH08]). In Anlehnung an die oben vorgestellten Punkte bedeutet dies nicht nur Ergebnisse, wie hier z.B. das Lernportal selbst, sondern vor allem Strukturen und Prozesse bei der Konzeption, Planung und Implementierung bei dem Aufbau eines Lernportals zu explizieren. Einige theoretische Aspekte der Qualitätssicherung und des Qualitätsmanagements spielen bei der Entwicklung des SuGI-Portals eine übergeordnete Rolle, so dass ihnen hier einige ausführliche, theoretische Vorüberlegungen zu widmen sind. Gemäß [GW07, Seite 24] ist die „Erstellung eines fertigen Produkts, das keiner Änderungen mehr bedarf (...), im Bereich von Lernportalen nicht realisierbar. Aufgrund kontinuierlicher Veränderung der Inhalte und eines ständig wachsenden Anspruchs an die Funktionalitäten erscheint die Entwicklung in Iterationen und Zyklen unabdingbar. Darüber hinaus ermöglicht dieses Vorgehen die Evolution ausgehend von einer ersten Version hin zu einem komplexen Portal unter Einbeziehung der Voraussetzungen, Bedürfnisse und Wünsche des realen Nutzers. (...) Zusätzlich wird eine Reaktion auf neue technische und funktionelle Anforderungen ermöglicht."

Lernportale – und als solches ist das SuGI-Portal zu sehen – stellen gemäß [Sch00, Seite 326] eine Sonderform von Portalen dar. Sie „sind elektronische Kundenschnittstellen im Internet, die dem Kunden Zugang zu Lerninhalten, Informationen und Bildungsmehrwertdiensten (z.B. Communities und Teletutoring) ermöglichen. Sie bilden das Web Front End für Content Management Systeme und Wissensdatenbanken".

[GW07] beschreiben ein generisch-iteratives Modell zur Qualitätssicherung bei der Entwicklung von Lernportalen, das sich „insbesondere an den übergeordneten Produkten, die dem Entstehungsprozess zugrunde liegen" orientiert (ibid.:23). Dieses fasst die Portal-

entwicklung als einen mehrstufigen Prozess auf, bestehend aus der „(1) Entwicklung eines Konzepts, (2) Implementierung eines Prototyps und (3) Einsatz einer Portalversion“ (ibid.), der in einem iterativen Prozess mehrere Zyklen (Generationen) durchläuft. Hierzu sei erwähnt, dass bei der Planung und Konzeption des SuGI-Portals die oben genannten Aspekte durchaus nicht nur berücksichtigt, sondern auch in einem entsprechenden Strategiepapier (D1.1) dargelegt und somit explizit gemacht wurden. Ein solcher Methodenansatz, der sich von der Entwicklung reiner Softwareprodukte deutlich unterscheidet, scheint notwendig, da Websysteme nach [GW07, Seite 15] „im Unterschied zu traditionellen Softwareprodukten (...) zum einen durch einen kontinuierlichen Wechsel der Informationsinhalte und zum anderen durch ein ständiges Anwachsen der Anforderungen gekennzeichnet“ sind. So ist z.B. der Dialog mit dem Nutzer des Lernportals, wie ihn u.a. eine Evaluierung in der weiter unten diskutierten Form darstellt, ein wesentlicher Bestandteil des Qualitätsmanagements. Die Nutzer deutlich stärker in die Produktentstehung miteinzubeziehen, wird z.B. auch von [Gin02] und [Arn05] gefordert. [ARH03] (zitiert nach [GW07, Seite 16]) befürworten „(...) eine wiederholbare, nachvollziehbare und dokumentierte Vorgehensweise, die reproduzierbare und qualitätsgesicherte Entwicklungen sicherstellt“.

Als Substrat aus den besprochenen Ansätzen ergeben sich einige zentrale Punkte für den Aufbau und die Entwicklung von Lernportalen: die Entwicklung einer Portalstrategie, die Formulierung eines Anforderungskatalogs, eine Fach- und Portalkonzeption, die Realisierung (begleitet von verschiedenen Schritten der Qualitätssicherung) sowie die Einführung und Weiterentwicklung des Portals (Einführung und Evolution), die Bestimmung der Zielgruppe(n), die ggf. außerhalb institutioneller Grenzen liegen können, sowie ein iterativer Charakter des Entwicklungsprozesses [GW07, Seite 16] und [AGR+08]. Somit erhält man einen Entwicklungsprozess, angelegt als Phasen- oder Generationenmodell und begleitet von Qualitätssicherung bzw. Qualitätsmanagement, also der Einspeisung der Evaluierungsergebnisse in den Entwicklungsprozess, wie von [GW07] vorgeschlagen, der zu einem deutlich höherem Nutzen der Evaluierungsergebnisse führt und eine wissenschaftlich fundierte Begleitung beim Portalaufbau ermöglicht.

3 Umsetzung der Konzeption und Implementierung des SuGI-Portals

Angelehnt an die beschriebenen Vorgehensweisen wurde im Rahmen des Strategiepapiers (D1.1) ein solcher evolutionärer Ablauf definiert und während des Projektes konsequent durchgeführt. Eine der wesentlichen Aufgaben eines Lernportals ist es, einen guten Zugang zu den bereitgestellten Material zu ermöglichen. Im SuGI-Portal wird dies zum einen durch zielgruppen- bzw. kategoriebasierte Zugriffe erreicht und zum anderen durch eine Suchfunktion. Die Menüpunkte des Portals spiegeln inhaltliche Kategorien wieder, so wird über den Punkt *Veranstaltungen* eine Übersichtsseite zu allen dokumentierten Veranstaltungen erreicht. Dieser Zugriffsweg eignet sich für Nutzer, die bereits über eine genaue Vorstellung des gesuchten Informationsmaterials verfügen. Eine Image Map auf der Startseite stellt Einstiegspunkte bereit, die sich an bestimmte Zielgruppen richten und deren zu erwartende Informationsbedürfnisse berücksichtigen. Dieser Zugriffsweg eignet sich für

Nutzer, die durch das Portal stöbern oder sich einen ersten Überblick über bestimmte Themenbereiche verschaffen möchten. Die Suchfunktion richtet sich dagegen eher an Nutzer, die sich Vorschläge zu bestimmten Themen machen lassen wollen. Damit Anwender einen möglichst großen Nutzen aus der Suchfunktion ziehen können, werden die Suchergebnisse so dargestellt, das der damit verbundene Inhalt schnell erfasst werden kann.

Hinter den ‚Zielgruppenschnittstelle' verbergen sich im Falle von SuGI ein Typo3-Content-Management-System und eine relationale Wissens- bzw. Inhaltedatenbank, die, als Open Source-Produkte realisiert, eine hohe Flexibilität aufweisen. Über das Web-Frontend wird den Zielgruppen nicht nur ein Zugang zu intern und extern produzierten Lerninhalten etc. geboten, vielmehr besteht darüber hinaus die Möglichkeit, Aufzeichnungen von Inhalten bzw. Inhalte von Präsenzschulungen zu archivieren und wiederholt abrufbar zu machen, eigene, selbst produzierte Inhalte einem größeren Publikum zur Verfügung zu stellen sowie einige Grid-Communities exemplarisch ausführlich vorzustellen und deren Arbeitsweise anschaulich zu beschreiben, um so das D-Grid-Portal entsprechend zu ergänzen und den Nutzern tiefergehende Einblicke in die angewandte Arbeit mit dem Grid zu ermöglichen.

Abbildung 2: Kurzdarstellung von Inhalten

Suchmaschinen liefern häufig lediglich Listen von Referenzen auf Web-Seiten oder Dokumente, die den Anwendern jedoch kaum eine Einschätzung des referenzierten Inhalts ermöglichen. Da im SuGI-Portal Inhalte durch Metadaten beschrieben sind, wurde diese Information genutzt, um Suchergebnisse leicht erfassbar darzustellen. Abbildung 2 zeigt, wie ein einzelnes Inhaltselement kompakt im SuGI-Portal dargestellt wird. Diese Form der Darstellung wird nicht nur für Suchergebnisse, sondern für jede Art von aufgelisteten Informationsmaterialien verwendet. Über die Referenz *Details* erhält der Anwender eine noch ausführlichere Übersicht zu dem jeweiligen Inhalt. Das Anzeigen relevanter Stellen in multimedialen Inhalten direkt aus der Suchfunktion heraus ist geplant. Verschiedene Treffer, die sich auf das gleiche Informationsmaterial beziehen, werden erkannt und zusammengefasst. Suchergebnisse werden anhand der Fundstelle (Schlagwort, Titel, sonstige Beschreibungen) der Suchbegriffe bewertet und entsprechend sortiert. Darüber hinaus sind einige der Metadaten (Autoren, Schlagwörter, Veranstaltung etc.) untereinander verlinkt, so dass leicht weitere Inhalte mit demselben Kriterium gefunden werden können.

Abbildung 3 vermittelt einen Eindruck davon, in welcher Weise sich beispielsweise die optischen Attribute des Portals aus den Nutzerfeedbacks von Generation 0 zu Generation 1 entwickelt haben. Zu diesen Änderungen gehören Design (z.B. Farbe, positionierung der Menüs), Nutzerführung, Struktur, Informationsgrad und Art der Darstellung der Inhalte (Farbkodierung von unterschiedlichen Schwierigkeitsgraden, ...) und vieles mehr.

Abbildung 3: Screenshot der 0. und der 1. Generation des Portals

Aus diesem Vorüberlegungen heraus wurde die Planung des Generationenmodells des SuGI-Portals entwickelt, die folgende Aufteilung vorsieht: Das Lernportal wird in drei Generationen erscheinen (vgl. Abbildung 3):

Generation 0 wurde kurzfristig entwickelt und ermöglichte eine schnelle Publizierung der bisher generierten Lehrinhalte und Materialien. Darüber hinaus konnten durch die schnelle Bereitstellung der Lösung auch frühzeitig generelle Probleme wie die Integration in bestehende D-Grid-Infrastrukturen und dergleichen erkannt und korrigiert werden. Die hieraus gewonnen Erkenntnisse wurden in die folgende Generation integriert um eine stetige Verbesserung zu erreichen. Die Generation 0 wurde Anfang des Jahres 2008 veröffentlicht.

Generation 1 lieferte einen nahezu vollständigen Funktionsumfang und implementierte bereits viele der geplanten Maßnahmen zur Steigerung der Benutzerfreundlichkeit wie zum Beispiel eine erweiterte Volltextsuche. Templates zur einfachen Integration und übersichtlichen Darstellung von Informationen wie Aufzeichnungen und Online-Modulen sind entwickelt und integriert. Generation 1 des Lernportals wurde Anfang 2009 veröffentlicht.

In *Generation 2* werden die Erkenntnisse, die durch das Feedback zu den vorhergehenden Generation gewonnen wurden, umgesetzt sowie verfeinerte Strukturen und Funktionen implementiert sein. Darüber hinaus wird die Nachhaltigkeit hier eine wichtige Rolle spielen. Dies impliziert Funktionen, die das einfache Einstellen weiterer Inhalte durch ausgewählte Nutzergruppen ermöglichen. Das Release der Generation 2 des Lernportals ist für Mitte 2009 geplant.

Im folgenden Abschnitt werden schließlich die Ergebnisse der Evaluierung des SuGI-Portals beschrieben.

4 Evaluierung

Das SuGI-Webportal und die Schulungen sind Bildungsmaßnahmen, deren Wirksamkeit, Qualität und Nutzen während der Projektlaufzeit laufend systematisch untersucht und evaluiert werden (formative Evaluierung). Ziel ist es, die Angebote im Rahmen der Schulungsinfrastruktur optimal an die Bedürfnisse der Zielgruppen anzupassen und systematisch weiterzuentwickeln. Der folgende Abschnitt fasst wesentliche Aspekte und Ergebnisse der Evaluierung zusammen.

Um den Erfolg der Schulungen sowie die Qualität und Akzeptanz des Lernportals im Rahmen des Projekts planmäßig und zielgerichtet aufzubauen und zu sichern, werden Maßstäbe und Zielvorgaben zur Bewertung der im Projektverlauf konzipierten Schulungsmaßnahmen benötigt. Ziel ist die Gewährleistung eines flüssigen Entwicklungsprozesses, in dem Fehlentwicklungen erkannt und durch korrigierende Maßnahmen behoben werden. Die Zielvorgaben, an denen der Erfolg des Projektverlaufs gemessen wird, sind: (1) Akzeptanz und Attraktivität des Schulungsangebots innerhalb der Zielgruppe, (2) Effektivität der Schulungsmaßnahmen, (3) Optimierung des Lernportals. Als Teil des Qualitätsmanagements ist die Evaluierung in den Prozess der Portalentwicklung eingebunden, wie von [GW07] beschrieben. Dies führt zu einer deutlich effektiveren Umsetzung der Evaluierungsergebnisse. Zusätzlich zu den Zielvorgaben werden auf einer praktischen Ebene auch, die Themenfelder der Softwareergonomie [Sch, Hel02], das Nutzerverhalten [e-t] sowie Aspekte der Wirkungsforschung [PB06, Ker01] berücksichtigt. Dies führt zu einem Methodenmix bei der Abfrage. Gemäß [GHLE07, Seite 24] ermöglicht erst ein fortgeschrittenes Projektstadium die Anwendung unterschiedlicher Methoden. So ist z.B. erst beim produktiven Einsatz einer Portalversion eine umfassende Benutzerbefragung möglich, deren Daten für eine Evaluierung sinnvoll genutzt werden können.

Im Hinblick auf die Evaluierung des SuGI-Portals besteht der Methodenmix aus einem Online-Fragebogen, qualitativen und quantitativen Nutzerfeedbacks (Rezensionen durch ausgewählte Fachleute sowie strukturierte Befragungen, die sowohl schriftlich als auch mündlich durchgeführt werden) und der anonymisierten Auswertung der Zugriffsdaten. Es handelt sich um eine Prozessevaluierung[3], da das Projekt noch nicht abgeschlossen ist und die Evaluierungsergebnisse in die Weiterentwicklung des SuGI-Portals einfließen werden. Die Evaluierung wird intern, also von SuGI selbst durchgeführt. Neben einer Legitimationsfunktion hat sie eine Kontroll- und Dialogfunktion, da Fragebögen, Befragungen etc. einen Dialog mit den Teilnehmern bilden. Zentrale Frage der Evaluierung ist es, ob und wie das Lernportal von der Zielgruppe angenommen wird und inwiefern es sich als gewinnbringende Informations- und Schulungsquelle entwickelt.

Die Auswertung der Evaluierung des SuGI-Portals bestätigt das durch die Evaluierung der Schulungsveranstaltungen gewonnene, durchweg positive Gesamtbild der Schulungsaktivitäten bzw. verstärkt dessen Tendenzen. Das SuGI-Portal wird von den Zielgruppen z.T. sehr gut angenommen, die angebotenen Inhalte finden das Interesse der Nutzer und der Aufbau des Portals bzw. dessen Bedienung werden als übersichtlich und intuitiv erlernbar

[3] Der Begriff ‚Prozessevaluierung' bezeichnet in diesem Kontext die Evaluierung eines in seiner Laufzeit befindlichen Projektes, deren Ergebnisse in die weitere Projektarbeit einfließen. Er steht somit im Gegensatz zu einer summativen ex-post Produktanalyse, die ein bereits abgeschlossenes Projekt bewertet.

bewertet. Auffällig ist, dass das Portal bzw. die darüber angebotenen Schulungsmaterialien zumindest von an den Evaluierungen beteiligten Personen bislang nur in sehr geringem Umfang für eigene Lehrveranstaltungen, wie z.B. die Aus- und Weiterbildung von Studierenden und Mitarbeitern, genutzt werden. Darüber hinaus wurden kleinere technische Probleme bemängelt und im Wesentlichen angeregt, an der weiteren Verbesserung der Inhalte (z.B. deren Tonqualität) zu arbeiten und mehr Inhalte zur Verfügung zu stellen, die sich als Grundlage für Schulungen bzw. die Aus- und Weiterbildung von Studierenden und Mitarbeitern eignen.

5 Ergebnis / Ausblick

Die speziellen Rahmenbedingungen, die aus dem D-Grid getrieben wurden und das iterative und rekursive Vorgehen haben in diesem Fall zu einer Reihe von Innovationen geführt. So unterscheidet sich das SuGI-Portal in einer Reihe von Kriterien von herkömmlichen Lernportalen. Die meisten Lernportale, wie Ilias (eLearning Portal der Universität zu Köln), Prodo (Fachhochschule Köln), E-Campus (JL Universität Gießen) etc. spiegeln die Lernsituation in beispielsweise Schulsystemen oder Universitäten wieder. Sie unterstützen den Lehrbetrieb im Rahmen eines über Jahrzehnte etablierten Systems, das durch feste Strukturen wie beispielsweise der Besuch von Lehrveranstaltungen mit begleitender Lektüre, vertiefende Tutorien etc. gekennzeichnet ist. Für die Betreuung dieser Aktivitäten stehen (oftmals grundfinanzierte) Mitarbeiter zur Verfügung. Auf diese Situation ausgerichtet stellen die meisten Lernportale personalisierbare Bereiche bereit, in denen Lernende eben diese Informationen vorfinden, in ihrer Terminplanung unterstützt werden, sowie oftmals Tests zur Kontrolle des Lernerfolges durchführen können. Dies führt zu einer effektiv unterstützten Form des Lernens, die jedoch mit einem erheblichen Betreuungsaufwand verbunden ist. Im D-Grid hingegen stehen Mitarbeiter nur für eine begrenzte Zeit zur Verfügung. Ferner wird in konzentrierter Form über die Projektlaufzeit eine große Menge an Informationen generiert, verarbeitet und veröffentlicht, die nach Ablauf der Projektlaufzeit ihre Bedeutung nicht verloren haben und an nachfolgende Wissenschaftler weiter gegeben werden sollen. So stellt das SuGI-Portal eher eine Mischung aus klassischen Lernportalen und Systemen wie YouTube [You] dar. YouTube ist ein System, welches inhaltlich durch die Community betrieben wird. Jeder ist in der Lage Inhalte anderen zur Verfügung zu stellen und die Kontrolle über die Inhalte erfolgt ebenfalls weitgehend über die Community. Der Zugriff auf die Informationen ist explizit auf eine große Menge anonymer Nutzer ausgerichtet. So bildet es in einer nicht unerheblichen Menge von Attributen die Anforderungen des D-Grids ab. YouTube ist allerdings auf Videos spezialisiert, die in bestimmten Formaten vorliegen müssen. Als Bildungsplattform für das D-Grid ist diese Funktionalität zu gering. Daher strebt das SuGI-Projekt eine passende Synergie dieser beiden, hier beschriebenen Lösungen an. Ergebnis dieser Bemühungen ist eine Portallösung, die es erlaubt, Inhalte von unterschiedlicher Art, wie beispielsweise Texte, Links, Videos unterschiedlicher Formate, stark oder schwach interaktive Lernmodule, generiert von verschiedenen Werkzeugen und Editoren, Übungssysteme und dergleichen viele mehr zu verwalten. Gleichzeitig geht der Zugriff auf die bereitgestellten Informationen ähnlich einfach

von statten, wie etwa bei YouTube. In Generation 2 des SuGI-Portals wird die Betreuung im Wesentlichen aus der Community getrieben stattfinden. Jede Institution kann ihre Inhalte ohne größeren Aufwand auf das Portal stellen. Somit fließt die investierte Arbeit, die mit der Ausrichtung von Workshops und Seminaren einhergeht, direkt in Lernmaterialien ein, die nachhaltig einer weit größeren Community für einen längeren Zeitraum sichtbar und verfügbar gemacht werden – es herrscht ein erheblich stärker ausgeprägter Investitionsschutz. Mitarbeitern von Rechenzentren, die es zeitlich nicht einrichten können zu vielen externen Veranstaltungen zu reisen, wird es ermöglicht sich die für sie interessanten Beiträge gezielt in ihrem Arbeitstempo und angepasst an ihre Arbeitsplanung anzusehen. Teilnehmer von Veranstaltungen können Beiträge bei Bedarf bequem an ihrem PC nachbereiten. Somit stellt das SuGI-Portal eine innovative, nachhaltige und skalierende Bildungsplattform dar, die auf den besonderen Anforderungen des D-Grid beruht und das Potenzial hat, nachhaltig weiter betrieben zu werden.

Literatur

[AGR+08] Viktor Achter, Claudia Gayer, Bernd Reuther, Marc Seifert, and Peter Zanger. Konzeptpapier Trainings- und Schulungsinfrastruktur. Deliverable 1.1, SuGI, 2008.

[ARH03] Michael Amberg, Ulrich Remus, and Jochen Holzner. Portal-Engineering – Anforderungen an die Entwicklung komplexer Unternehmensportale. pages 795–818, 2003.

[Arn05] Henrik Arndt. Anforderungen an einen spezifischen Entwicklungsprozess hochfunktioneller Websites. In Andreas Auinger, editor, *Workshop-Proceedings der 5. fachübergreifenden Konferenz Mensch und Computer*, pages 47–51, Wien, 2005.

[Bal98] Helmut Balzert. *Lehrbuch der Software-Technik - Software-Management, Software-Qualitaetssicherung, Unternehmensmodellierung*. Spektrum, 1998.

[BH08] Florian Buch and Yorck Hener. Evaluation des Bildungsportals Sachsen. Arbeitspapier 80, Centrum für Hochschulentwicklung, Gütersloh, 2008.

[BRNP04] Monika Bias, Konrad Ringel, Alfred Nagel, and Christian Priller. Qualitätsmanagement für kleine und mittlere Unternehmen: Leitfaden zur Einführung und Weiterentwicklung eines Qualitätsmanagementsystems nach der Normenreihe DIN EN ISO 9000:2000. Technical report, München, 2004.

[DGR] D-Grid - http://www.d-grid.de.

[DL00] Klaus Doppler and Christoph Lauterburg. *Change Management. Den Unternehmenswandel gestalten.* Campus Fachbuch, January 2000.

[Dör03] Jana Dörfel. Virtuell studieren in Deutschland – Aktueller Stand und Entwicklungstendenzen. Diplomarbeit, Hochschule für Technik und Wirtschaft Dresden (FH), Fachbereich Informatik/Mathematik, Dresden, 2003.

[e-t] e-teaching.org – Usability – http://www.e-teaching.org/.

[GHLE07] Birgit Gaiser, Friedrich W. Hesse, and Monika Lütke-Entrup, editors. *Bildungsportale. Potenziale und Perspektiven netzbasierter Bildungsressourcen*. Oldenbourg, München, 2007.

[Gin02] Athula Ginige. Web engineering: managing the complexity of web systems development. In *SEKE*, pages 721–729, 2002.

[Gro04] The Standish Group. The CHAOS Report - The Standish Group, 2004.

[GW07] Birgit Gaiser and Benita Werner. Qualitätssicherung beim Aufbau und Betrieb eines Bildungsportals. In Birgit Gaiser, Friedrich W. Hesse, and Monika Lütke-Entrup, editors, *Bildungsportale. Potenziale und Perspektiven netzbasierter Bildungsressourcen*, pages 13–28. Oldenbourg, München, 2007.

[Hel02] Günter Hellbardt. Software-Ergonomie: Material zur Vorlesung, 2002.

[Hof95] Karl-Heinz Hoffmann. *Transparenz, Evaluation und Qualitätssicherung: Lehre auf dem Prüfstand*, pages 137–147. Bertelsmann-Stiftung, Gütersloh, 1995.

[Ker01] Michael Kerres. *Multimediale und telemediale Lernumgebungen: Konzeption undEntwicklung*. Oldenbourg, München [u.a.], 2., vollst. überarb. aufl. edition, 2001.

[PB06] Annabell Preussler and Peter Baumgartner. *Qualitätssicherung in mediengestützten Lernprozessen: Sind theoretische Konstrukte messbar?*, pages 73–85. Number 36 in Medien in der Wissenschaft. Waxmann, Münster, 2006.

[Sch] Ursula Schulz. Web Usability – http://www.bui.haw-hamburg.de/pers/ursula.schulz/webusability/webusability.html.

[Sch00] Susanne Schestak. *Bildungsportale: Neue Zugänge zu Wissen*, pages 325–329. Number 10 in Medien in der Wissenschaft. Waxmann, Münster, 2000.

[Wal95] Ernest Wallmüller. *Ganzheitliches Qualitätsmanagement in der Informationsverarbeitung*. Hanser, 1995.

[You] YouTube – http://de.youtube.com/.

F&L-Grid: Eine generische Backup und Recovery Infrastruktur für das D-Grid

Markus Mathes, Steffen Heinzl, Roland Schwarzkopf, Bernd Freisleben
Fachbereich Mathematik und Informatik, Philipps-Universität Marburg
Hans-Meerwein-Str. 3, 35032 Marburg

{mathes,heinzl,rschwarzkopf,freisleb}
@informatik.uni.marburg.de

Abstract: Grid Computing wird oftmals zur Durchführung zeitintensiver Experimente, die eine enorme Menge an Daten produzieren, verwendet. Da existierende Backup und Recovery Lösungen basierend auf GridFTP oder RFT detaillierte technische Kenntnisse bezüglich Konfiguration und Nutzung erfordern, sind diese nicht unbedingt für alle Anwender geeignet. Viele Wissenschaftler bevorzugen eine möglichst einfache und bedienungsfreundliche Lösung, um ihre experimentellen Ergebnisse zu sichern.

Das F&L-Grid Projekt, welches die Entwicklung eines Grids für Forschung und Lehre beabsichtigt, ist ein Teil des D-Grid. Hauptziel von F&L-Grid ist der Entwurf und die Entwicklung einer generischen Backup und Recovery Infrastruktur für beliebige Grid-Umgebungen. Dieser Beitrag diskutiert den aktuellen Projektstatus von F&L-Grid und skizziert den Entwurf und die Implementierung der generischen Backup und Recovery Infrastruktur.

1 Einleitung

Grid Computing Umgebungen sind heterogene Sammlungen von Hard- und Software, die sich an verschiedenen Orten befinden und von verschiedenen Organisationen zur Verfügung gestellt werden. Die Hauptziele solcher Umgebungen sind die gemeinsame Nutzung von Ressourcen und Lösung von Problemen über die Grenzen von individuellen Institutionen hinweg [FKT01]. Um den Benutzern einen bequemen Zugriff auf Ressourcen über standardisierte Schnittstellen zu ermöglichen, verwendet man service-orientierte Grid Middleware basierend auf dem Web Service Resource Framework (WSRF) [OAS]. Beispiele für solche service-orientierte Grid Middleware sind das Globus Toolkit 4.x [Pet08] und Unicore/GS [Rom99]. Normalerweise unterstützt eine solche Middleware Laufzeitkomponenten, Ausführungs- und Informationsmanagement, Sicherheit und Datenhaltung. Unter Verwendung einer service-orientierten Grid Middleware werden Applikationen aus mehreren Grid Services komponiert. Ein Grid Service implementiert dabei einen kleinen Teil der gesamten Funktionalität. Große Applikationen werden in eine Vielzahl von Grid Services zerlegt, die dann zu neuen Applikationen komponiert werden können. Diese Vorgehensweise vermindert die redundante Implementierung von Funktionalität und erhöht gleichzeitig die Flexibilität.

Auf Grund ihrer Rechenleistung werden Grid Computing Umgebungen oftmals verwen-

det, um komplexe Experimente durchzuführen, z.B. innerhalb der Hochenergiephysik. Solche Experimente produzieren üblicherweise eine große Menge an Daten. Zum Zwecke der Datenhandhabung bieten viele service-orientierte Grid Middleware Systeme (funktional sehr eingeschränkte) Tools an. Beispielsweise bietet das Globus Toolkit GridFTP – eine Kommandozeilen-basierte FTP-Lösung für Grid-Umgebungen – und Reliable File Transfer (RFT) – einen Service-Wrapper für GridFTP. Diese mitgelieferten Tools werden häufig zum Backup und Recovery zweckentfremdet. Leider sind sie oftmals schwer zu bedienen und überfordern Wissenschaftler, die lediglich ihre Daten sichern wollen, ohne dabei die Interna der Grid Middleware zu kennen.

In diesem Beitrag wird eine generische Backup und Recovery Infrasktruktur präsentiert, die innerhalb des F&L-Grid Projektes entwickelt wurde. F&L-Grid ist Teil der D-Grid Initiative [NKG07] und wird von folgenden Projektpartnern getragen: T-Systems SfR und Karlsruhe Institute of Technology als Dienstanbieter, DFN-Verein als Anbieter der Netzwerkinfrastruktur und Philipps-Universität Marburg als Entwickler.

Im diesem Beitrag wird auf folgende Themen eingegangen:

- Die generellen Anforderungen an eine Backup und Recovery Infrastruktur für Grid Umgebungen werden identifiziert.
- Basierend auf den Anforderungen wird eine einfach zu verwendende Backup und Recovery Infrastruktur für beliebige Grid Umgebungen vorgestellt. Wissenschaftlern wird es ermöglicht, ihre Daten einfach und ohne das Eingreifen eines Administrators zu sichern und wiederherzustellen.
- Um Ausfallsicherheit zu garantieren, setzt die F&L-Grid Lösung auf einem kommerziellen Backup und Recovery Backend auf. Die Details der kommerziellen Lösung werden jedoch vor dem Benutzer verborgen.

Der Rest dieses Beitrags ist wie folgt organisiert: In Abschnitt 2 werden die Anforderungen an die F&L-Grid Backup und Recovery Lösung diskutiert und die entwickelte Architektur präsentiert. Abschnitt 3 präsentiert Implementierungsdetails. Weitere Backup und Recovery Lösungen für Grid-Umgebungen werden in Abschnitt 4 diskutiert. Abschnitt 5 fasst den gesamten Beitrag zusammen und gibt einen Ausblick auf zukünftige Entwicklungen.

2 Entwurf einer generischen Backup und Recovery Infrastruktur

Die Deutsche Grid-Initiative (D-Grid) [NKG07] wurde 2003 als Teil der nationalen e-Science Initiative des Bundesministeriums für Bildung und Forschung (BMBF) [BMB] gegründet. Ihr Ziel ist der Aufbau einer nachhaltigen Grid-Infrastruktur in Deutschland, um die Voraussetzungen für e-Science zu schaffen. D-Grid besteht aus mehreren Community-Projekten und dem D-Grid-Integrationsprojekt.

Das in diesem Beitrag vorgestellte F&L-Grid Projekt hat zum Ziel, ein generisches, service-orientiertes Grid für Forschung und Lehre aufzubauen. Die angebotenen Dienste werden

auf der Infrastruktur des DFN [DFN] bereitgestellt, die die Universitäten und Forschungseinrichtungen in Deutschland miteinander verbindet. In der aktuellen Projektphase wird ein Backup und Recovery Dienst entwickelt. Im Rahmen dieses Dienstes agieren einige Forschungseinrichtungen als Anbieter, andere als Benutzer, während das DFN die Rolle des Dienstvermittlers zwischen den beteiligten Einrichtungen übernimmt. Eine mögliche Erweiterung, die beispielsweise in einem Anschlussprojekt realisiert werden könnte, ist die Erweiterung um eine Archivierungsfunktion.

In den folgenden Abschnitten werden die Anforderungen von Anbietern und Benutzern dargestellt, pull- und push-basierte sowie Knoten- und Benutzer-basierte Ansätze zum Backup und Recovery verglichen, verschiedene Backup-Strategien erörtert und ein Überblick über die F&L-Grid Architektur gegeben.

2.1 Anforderungsanalyse

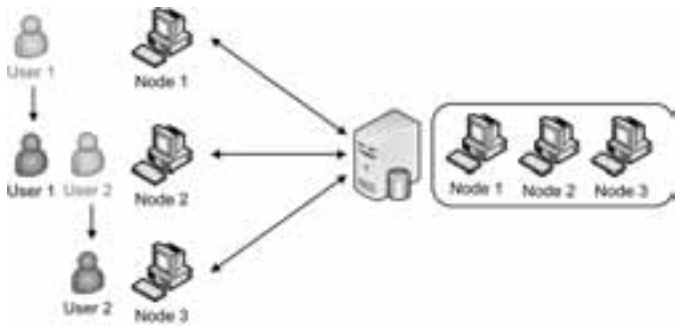
Die entscheidenden Anforderungen, die von F&L-Grid Anbietern und Benutzern gestellt wurden, sind:

- **Minimalinvasivität:** Der Backup und Recovery Dienst darf keine bereits existierenden betrieblichen Abläufe der Anbieter beeinflussen.
- **leichte Verwendbarkeit:** Es soll Wissenschaftlern ermöglicht werden, experimentelle Ergebnisse zu sichern oder wiederherzustellen, ohne dass Hilfe vom Administrator oder Help-Desk in Anspruch genommen werden muss, was bei kommerziellen Lösungen oft nötig ist.
- **einfache Installation:** Die notwendige Software soll sowohl auf Anbieter- als auch auf Benutzerseite einfach installiert werden können. Insbesondere Aktualisierungen der Client-Software sollen auf den Maschinen der Benutzer automatisch eingespielt werden.
- **Nachhaltigkeit:** Der entwickelte Backup und Recovery Dienst soll langfristig, also auch nach Abschluss des Projekts, vom DFN und den Anbietern angeboten werden können.
- **austauschbares Backend:** Eine Abhängigkeit des Dienst-Backends von einem speziellen kommerziellen System zur Backup und Recovery soll vermieden werden, so dass ein zukünftiger Austausch dieses Systems möglich ist.
- **Betriebssystemunabhängigkeit:** Es muss möglich sein, den Backup und Recovery Dienst auf verschiedenen Betriebssystemen zu benutzen, ohne dass verschiedene Versionen des Dienstes und der Client-Software gepflegt werden müssen.
- **Unterstützung verschiedener Schnittstellen:** Der Backup und Recovery Dienst soll sowohl innerhalb als auch außerhalb von Grids verwendbar sein.

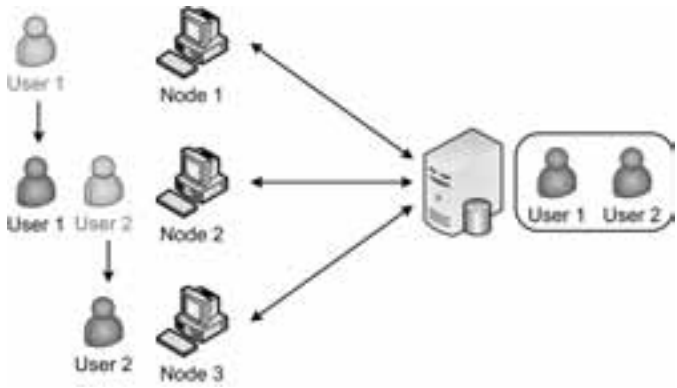
2.2 Pull- vs. Push-basierte Backup und Recovery Ansätze

Es gibt zwei Möglichkeiten, ein Backup durchzuführen: pull- und push-basiert. Beim *pull-basierten* Ansatz wird das Backup von der Server-Seite der Backup-Lösung gestartet. Dabei kommt meist ein voreingestellter Zeitplan zum Einsatz, der die Backups zu Zeiten mit niedriger Auslastung durchführt. Beim *push-basierten* Ansatz wird das Backup, z.B. nach dem Abschluss eines Experiments, vom Benutzer manuell angestoßen.

Da eine push-basierte Client-Software ohne Eingriff eines Administrators installiert und aktualisiert werden kann, wurde dieser Ansatz von den Projektpartnern gewählt.



(a) Knoten-basiertes Backup und Recovery.



(b) Benutzer-basiertes Backup and Recovery.

Abbildung 1: Beispiel eines Knoten- und Benutzer-basierten Backup und Recovery Ansatzes.

2.3 Knoten- vs. Benutzer-basierte Backup und Recovery Ansätze

Viele kommerzielle Lösungen unterstützen nur einen *Knoten-basierten* Ansatz für Backup und Recovery, weil sich die Dateisysteme auf unterschiedlichen Knoten erheblich unterscheiden können. Allerdings hat der Knoten-basierte Ansatz einen wesentlichen Nachteil:

ein Benutzer, der auf mehreren Knoten arbeitet, kann jeweils nur die Daten des gerade von ihm benutzten Knotens wiederherstellen, aber nicht alle seine Daten. Um dieses Problem zu lösen, muss ein *Benutzer-basierter* Ansatz gewählt werden, der eine Knoten-übergreifende Wiederherstellung von Daten ermöglicht.

Abbildung 1(a) zeigt einen Knoten-basierten Backup und Recovery Ansatz. Benutzer 1 und Benutzer 2 können ihre Daten nicht wiederherstellen, nachdem sie von Knoten 1 zu Knoten 2 bzw. von Knoten 2 zu Knoten 3 gewechselt sind. Mit dem Benutzer-basierten Ansatz wird eine Wiederherstellung von Daten auch nach einem Wechsel des Knotens möglich, wie Abbildung 1(b) zeigt.

Im Rahmen des F&L-Grid Projekts wird der Benutzer-basierte Backup und Recovery Ansatz realisiert.

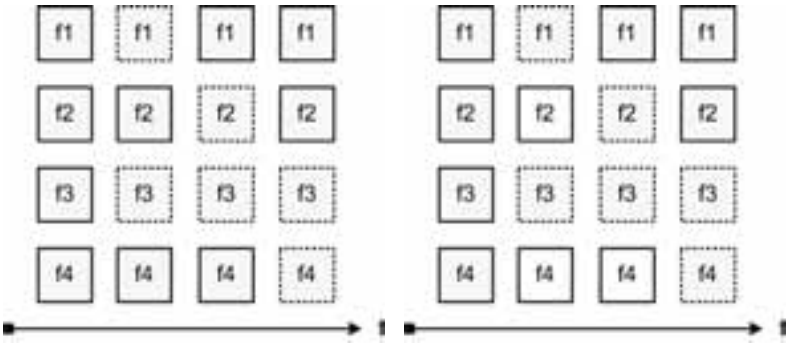
2.4 Inkrementelles, differentielles und Voll-Backup

Man unterscheidet zwischen 3 Backup-Strategien: inkrementelles, differentielles und vollständiges Backup. Ein *Voll-Backup* enthält alle ausgewählten Dateien. Ein *differentielles Backup* enthält alle Dateien, die seit dem letzten Voll-Backup geändert wurden. Ein *inkrementelles Backup* enthält nur die Dateien, die seit dem letzten inkrementellen Backup oder Voll-Backup geändert wurden.

In den Abbildungen 2(a), (b) und (c) werden Beispiele für ein Voll-Backup sowie ein inkrementelles und differentielles Backup von vier Dateien gegeben. Die Rechtecke stellen Dateien dar. Gestrichelte Linien zeigen eine Änderung seit dem letzten Backup an, durchgehende Linien stehen für unveränderte Dateien. Eine (gelbe) Füllung eines Rechtecks zeigt an, dass die Datei im jeweiligen Backup enthalten ist, wohingegen ungefüllte Rechtecke Dateien anzeigen, die nicht enthalten sind.

Der größte Vorteil der inkrementellen Backup-Strategie ist eine kürzere Backup-Dauer und geringerer Speicherverbrauch, weil nur die Änderungen seit dem letzten Backup enthalten sind. Um die Dateien wiederherzustellen, müssen das letzte Voll-Backup und alle danach durchgeführten inkrementellen Backups wiederhergestellt werden, was zu einer langen Recovery-Dauer führt. Die Verwendung der differentiellen Backup-Strategie verkürzt die Recovery-Dauer, weil nach dem Voll-Backup nur das letzte differentielle Backup wiederhergestellt werden muss. Dieser Vorteil geht jedoch zu Lasten der Backup-Dauer und des Speicherverbrauchs.

Im Rahmen des F&L-Grid Projekts wurde von den Partnern die Voll-Backup Strategie aus der Sicht des Fat-Clients (siehe Abschnitt 2.6) ausgewählt. Das ist nötig, weil der Fat-Client unter anderem als Zwischenspeicher fungiert und die Benutzerdaten nur temporär speichert, was differentielle oder inkrementelle Backups bei verschiedenen Backup und Recovery Lösungen ausschließt. Aus Sicht des Benutzers wird eine inkrementelle Backup-Strategie umgesetzt, da immer nur die seit dem letzten Backup modifizierten Daten zum Fat-Client übertragen werden.



(a) Voll-Backup. (b) Differentielles Backup.

(c) Inkrementelles Backup.

Abbildung 2: Beispiele für die unterschiedlichen Backup-Strategien (gestrichelte Rechtecke sind nach dem letzten Backup modifizierte Dateien, eingefärbte Rechtecke sind im aktuellen Backup enthaltene Dateien).

2.5 Metadaten

Die Metadaten zu den Dateien (z.B. Zugriffsrechte, Zeitstempel, etc.) unterscheiden sich auf den verschiedenen Dateisystemen (z.B. NTFS, ReiserFS, ZFS) und Betriebssystemen (z.B. Windows, Linux, MacOS). Folglich ist eine einheitliche Repräsentation dieser Metadaten ein schwieriges Unterfangen.

In F&L-Grid wird eine Schnittmenge von Metadaten verwendet (z.B. Datei-/Verzeichnisname, Zeitpunkt der Erzeugung und letzten Änderung, Dateigröße) die auf möglichst viele Betriebssysteme übertragbar ist. Alle relevanten Metadaten werden in einer Datenbank auf dem Fat-Client gespeichert (siehe Abschnitt 2.6).

2.6 Architekturskizze

Die Architekturskizze in Abbildung 3 spiegelt die identifizierten Anforderungen wider.

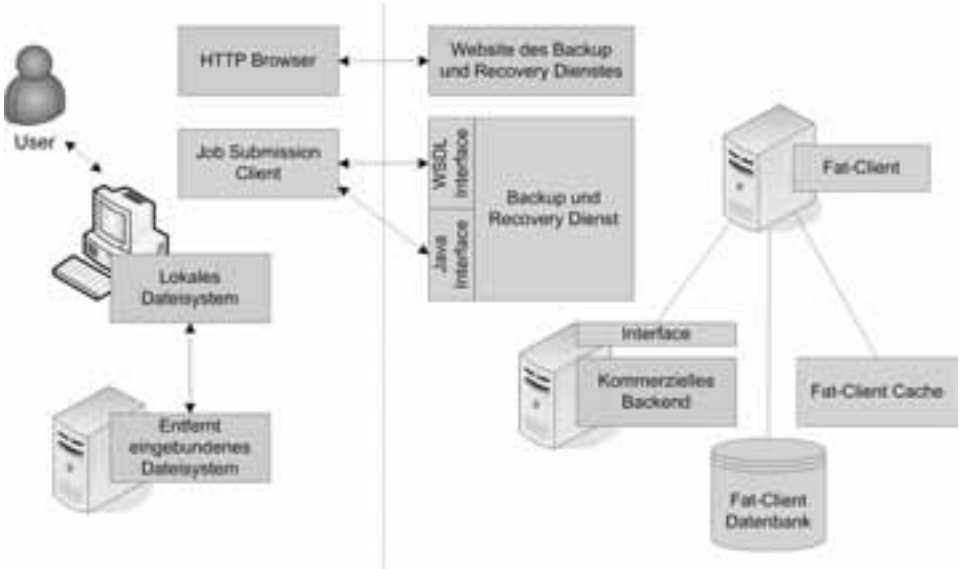


Abbildung 3: Architektur des F&L-Grid Backup und Recovery Dienstes.

Auf der Benutzer-Seite wird der sogenannte *Job Submission Client* benötigt, um auf die Backup und Recovery Infrastruktur zuzugreifen. Diese Software wird auf einer Website angeboten und kann mit dem Browser heruntergeladen werden. Damit kann der Benutzer jedes lokale oder per Netzwerk eingebundene Laufwerk sichern.

Die wichtigste Komponente auf der Anbieter-Seite ist der sogenannte *Fat-Client*. Er ermöglicht einen einfachen Zugriff auf die Backup und Recovery Infrastruktur und versteckt dabei viele Details vor dem Benutzer. Dabei übernimmt er die Kommunikation mit dem Backup und Recovery Backend, verwaltet zum Backup gehörige Daten in einer speziellen Datenbank und pflegt seinen Cache. Als Backup und Recovery Backend kann eine beliebige kommerzielle Lösung eingesetzt werden, da die Schnittstelle beliebig austauschbar ist. In der Datenbank werden die Metadaten und abrechnungsrelevanten Informationen, z.B. wie viele Backups ein bestimmter User ausgeführt hat, gespeichert. Der Cache wird benutzt, um die zu einem Backup oder Recovery Vorgang gehörenden Daten bis zu dessen Abschluss vorzuhalten. Das kommerzielle Backend liest beim Backup aus diesem Cache und schreibt beim Recovery in ihn. Diese Abläufe werden vom Backup und Recovery Dienst koordiniert.

2.7 Backup

Ein Backup-Vorgang besteht aus sechs aufeinander folgenden Schritten, die in Abbildung 4 dargestellt sind.

Abbildung 4: Bearbeitung eines Backup-Vorgangs.

1. **Authentication/Authorization:** Der Benutzer, der einen Backup-Vorgang anstoßen will, wird identifiziert, und der Zugriff auf das System wird ihm gestattet.

2. **Scan:** Die Metadaten von allen für das Backup ausgewählten Dateien/Verzeichnissen werden eingelesen und zu einer Liste zusammengefasst.

3. **Intersection:** Um unnötiges Kopieren von Daten zwischen dem Knoten des Benutzers und dem Fat-Client zu vermeiden, werden in diesem Schritt alle Dateien/Verzeichnisse von der Liste entfernt, die sich seit dem letzten Backup nicht mehr geändert haben. Dazu werden die Informationen über bereits gesicherte Dateien aus der Datenbank verwendet.

4. **Copy:** Alle noch auf der Liste stehenden Dateien/Verzeichnisse werden vom Knoten des Benutzers über eine gesicherte Verbindung in den Cache des Fat-Client kopiert.

5. **Backup:** Die kommerzielle Backup und Recovery Lösung wird verwendet, um die Dateien im Cache des Fat-Client zu sichern. Danach werden die Informationen über gesicherte Dateien/Verzeichnisse in der Datenbank aktualisiert.

6. **Cleanup:** Im letzten Schritt werden die Dateien dieses Backup-Vorgangs wieder aus dem Cache gelöscht, um Platz für folgende Backup und Recovery Vorgänge zu schaffen.

3 Implementierung

Dieser Abschnitt gibt einen Überblick der ausgewählten Technologien zur Implementierung des Backup und Recovery Dienstes, erklärt, wie auf den Backup und Recovery Dienst zugegriffen werden kann und beschreibt die Verarbeitung eines Backup Jobs im Detail.

3.1 Auswahl geeigneter Technologien

Basierend auf den identifizierten Anforderungen wurden die folgenden Technologien zur Implementierung des Backup und Recovery Dienstes ausgewählt.

Um *Minimalinvasivität* zu garantieren, wurde Java Web Start für die Implementierung des Job Submission Client gewählt. Eine Java Runtime Environment (JRE) [Mic] ist heute auf fast allen Computern verfügbar, so dass die Installation zusätzlicher Software vermieden werden kann. Außerdem sichert die Verwendung von Java die *Unabhängigkeit vom verwendeten Betriebssystem*. Mit Java Web Start wird eine neue Version der Software bezogen, sobald diese zur Verfügung steht. Außerdem erlaubt Java Web Start die Signatur der Software zu einem vertrauenswürdigen Zertifikatsherausgeber zurück zu verfolgen. Diese Funktionalitäten ermöglichen einen *einfachen Software Roll-out*.

Durch die Verwendung generischer Java Interfaces wird es möglich, verschiedene kommerzielle Backup und Recovery Lösungen als Backend einzusetzen (z.B. IBM Tivoli Storage Manager, EMC NetWorker), was den *einfachen Austausch des Backup und Recovery Backend* ermöglicht. Eine klar strukturierte und übersichtliche GUI ermöglicht dem Benutzer eine *einfache Bedienung*. Um einen Zugriff auf den Backup und Recovery Dienst über verschiedene Schnittstellen zu ermöglichen, wurde eine generische Schnittstelle definiert, aus der beliebige konkrete Schnittstellen abgeleitet werden können, z.B. eine Beschreibung in der Web Service Description Language (WSDL) [W3C06]. Um die *Nachhaltigkeit* des angebotenen Dienstes zu garantieren, hat der DFN-Verein ein Geschäftsmodell entwickelt, das die Rechte und Pflichten auf Nutzer- und Anbieterseite regelt.

In einer Umfrage, welche durch den ZKI Arbeitskreis Netzwerkdienste [ZKI] durchgeführt wurde, wurden 41 wissenschaftliche Einrichtungen in Deutschland nach deren Backup und Recovery Software befragt. Da 66% dieser Institutionen bereits IBM Tivoli Storage Manager (TSM) verwenden, basiert die prototypische Implementierung des Backup und Recovery Dienstes ebenfalls auf IBM TSM als Backend. Da IBM TSM lediglich Knoten-basiertes Backup/Recovery ermöglicht, bietet unsere Implementierung zusätzliche Funktionen für Benutzer-basiertes Backup/Recovery. Alle Informationen für ein Benutzer-basiertes Backup/Recovery sowie Accounting und Billing werden in der Fat-Client Datenbank gespeichert. Folglich bietet die Fat-Client Datenbank einen globalen Blick auf den gesamten Backup und Recovery Dienst.

3.2 Zugriff auf den Backup und Recovery Dienst

Ein Benutzer des Backup und Recovery Dienstes benötigt den Job Submission Client, um auf den Dienst zugreifen zu können, d.h. um einen Backup oder Recovery Job abzusetzen. Hierzu lädt der Benutzer den Job Submission Client in Form einer Java Web Start Anwendung von einer Website herunter. Der Benutzer muss den Job Submission Client nur einmal manuell herunterladen. Anschließend werden neue Version automatisch durch die Java Web Start Technologie heruntergeladen.

Um die Software herunterladen zu können, muss sich der Benutzer gegenüber seiner Hei-

matorganisation via Shibboleth [SJWA06] authentifizieren (*Authentication/Authorization Phase*). Beispielsweise kann sich der Benutzer mittels Benutzername und Passwort beim Identity Provider seiner Heimateinrichtung authentifizieren. Shibboleth wurde verwendet, da es sich sehr gut in das existierende Service-Portfolio des DFN, welches ebenfalls Shibboleth benutzt, integriert.

3.3 Absetzen eines Backup-Jobs

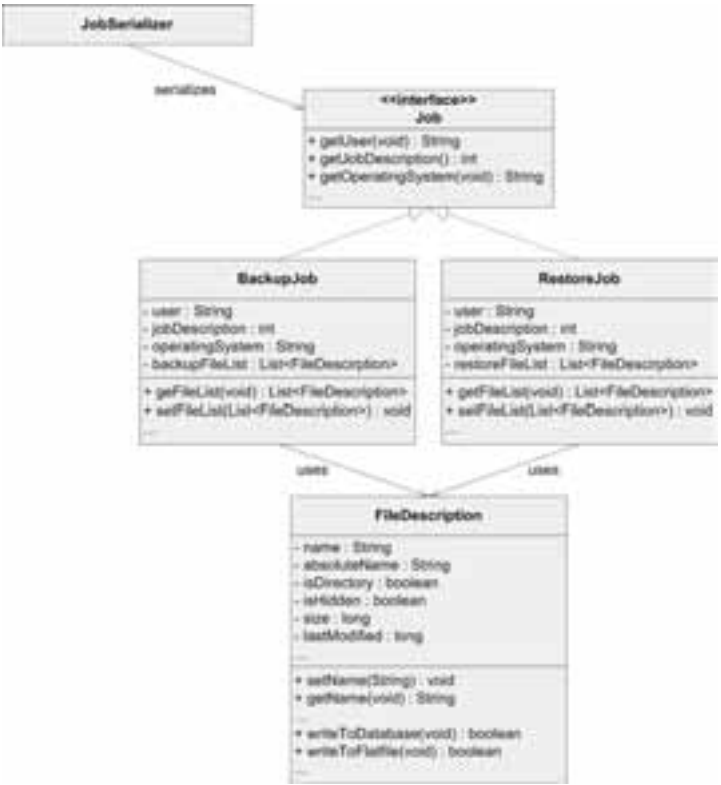


Abbildung 5: Klassenhierarchie `BackupJob` und `RecoveryJob`.

Ein Backup-Job wird durch die Auswahl aller relevanten Dateien und Verzeichnisse mit Hilfe des Job Submission Client definiert (*Scan Phase*). Der Backup-Job beschreibt die Dateien und Verzeichnisse durch deren Namen, den Hostnamen, von dem das Backup erfolgt, den Benutzernamen, den Zeitstempel der letzten Änderung und die Zugriffsrechte. Dieses `Job`-Objekt kann dann zum Backup und Recovery Dienst in Form eines Grid Service Aufrufs oder als ein serialisiertes Java-Objekt gesendet werden, je nach verwendetem Interface. Der Dienst entfernt alle Dateien/Verzeichnisse von der Liste, die seit dem letzten Backup nicht modifiziert wurden (*Intersection Phase*). Dies kann durch eine Anfrage bei

der Fat-Client Datenbank mittels JDBC ermittelt werden. Die modifizierte Liste wird an den Job Submission Client zurückgesendet, der die Daten anschließend mit Hilfe des Secure Copy Protocol (SCP) über eine Secure Shell (SSH) Sitzung an den Fat-Client sendet (*Copy Phase*). Der Prototyp verwendet die Java Secure Channel (JSch) [JSc] Bibliothek, welche eine Implementierung von SSH und SCP anbietet. Nachdem die Daten vollständig übertragen wurden, wird ein TSM Kommandozeilen-Client auf Seite des Fat-Client gestartet, um die Daten ins Backend zu übertragen (*Backup Phase*). Sobald das Backup erfolgreich durchgeführt wurde, werden Informationen zum Backup-Job in der Fat-Client Datenbank hinterlegt und die kopierten Daten werden aus dem Fat-Client Cache entfernt (*Cleanup Phase*). Die Zugriffsrechte werden für jede Datei als String hinterlegt. Derzeit werden UNIX Zugriffsrechte und Windows NTFS Zugriffsrechte mit Hilfe des Windows-Tools *cacls* (change access control lists) unterstützt.

Die Klassenhierarchie von Backup und Recovery Jobs wird in Abbildung 5 gezeigt. Die wesentlichen Eigenschaften eines `BackupJob` und eines `RecoveryJob` sind im Interface `Job` gekapselt. Beide spezialisierten `Job`-Klassen benutzen eine `FileDescription`, um relevante Dateien und Verzeichnisse zu beschreiben. Der `JobSerializer` wird benutzt, um einen Grid Service Aufruf oder ein serialisiertes Java-Objekt zu erzeugen.

4 Verwandte Arbeiten

Derzeit existieren nach unserem Wissen keine generischen und einfach verwendbaren Backup und Recovery Lösungen für Grid-Umgebungen. Folglich beschränken sich die verwandten Arbeiten auf Standardtechnologien für Datentransport in Grid-Umgebungen: GridFTP, RFT, RLS und OGSA-DAI.

Aktuelle service-orientierte Grid Middleware bietet von Haus aus Funktionalität zur Verwaltung von Daten. Basierend auf dieser Funktionalität wurden in einigen Forschungsprojekten oftmals proprietäre Backup und Recovery Lösungen entwickelt. Das Globus Toolkit 4.x [Pet08] beispielsweise bietet Funktionen zur Datenübertragung (GridFTP und Reliable File Transfer (RFT)) und Datenreplikation (Replica Location Service (RLS)). GridFTP ist eine für Grid-Umgebungen optimierte Version des weitverbreiteten File Transfer Protocol (FTP) und bietet eine effiziente, sichere und robuste Übertragung von Daten. Das Globus Toolkit beinhaltet einen GridFTP Server (`globus-gridftp-server`) und einen GridFTP Kommandozeilen-Client (`globus-url-copy`), um Daten bereitzustellen und zu übertragen. RFT bietet eine service-orientierte Schnittstelle zu GridFTP und ermöglicht zusätzlich das Scheduling von Datenübertragungsjobs. Die Replikation von Daten innerhalb eines Grids verbessert die Zugriffseffizienz. Um Replikate zu verwalten, bietet das Globus Toolkit RLS – eine einfache Registry für selbige. Die Verwendung von Middleware-spezifischer Funktionalität zur Datenverwaltung führt zu zwei Hauptproblemen:

- Die Backup und Recovery Lösung kann nur mit Aufwand wiederverwendet werden, da sie an die Middleware gebunden ist.

- Detailwissen über die Interna der Middleware ist erforderlich. Dies überfordert jedoch oftmals einfache Benutzer und macht das Eingreifen eines Administrators notwendig.

OGSA-DAI [ACHH+07] bietet einen Dienst, um auf Daten zuzugreifen, die aus verschiedenen Quellen stammen, z.B. Datenbanken oder flache Dateien. Außerdem bietet OGSA-DAI Funktionen, um Daten abzufragen, zu transformieren und auf verschiedene Weise auszuliefern. OGSA-DAI kann zwar verwendet werden, um Kopien von Daten anzulegen, jedoch ist die Durchführung zuverlässiger Backups damit nur schwierig möglich.

5 Zusammenfassung und Ausblick

In diesem Beitrag wurde eine generische Infrastruktur für Backup und Recovery in Grid-Umgebungen präsentiert.

Im F&L-Grid Projekt als Teil der D-Grid Initiative wurden zunächst die Anforderungen für eine solche, generische Backup und Recovery Infrastruktur identifiziert, und anschließend wurde eine geeignete Architektur entworfen. Der Backup und Recovery Dienst ist einfach zu benutzen und kann in beliebigen service-orientierten Grid-Umgebungen eingesetzt werden. Wissenschaftlern wird es ermöglicht, ihre Daten schnell, einfach und ohne das Eingreifen eines Administrator zu sichern und wiederherzustellen. Als Backend kann eine beliebige, kommerzielle Backup und Recovery Lösung eingesetzt werden, deren Details vor dem Anwender versteckt werden.

Zukünftig sollen weitere Metadaten für Dateien und Verzeichnisse unterstützt werden, z.B. die Vererbung von Zugriffsrechten. Außerdem wird eine Accounting und Billing Komponente entwickelt werden, welche Informationen aus der Fat-Client Datenbank verwendet, um automatisch Backup/Recovery Jobs abrechnen zu können. Ebenfalls von Bedeutung ist die Durchführung von Performance- und Skalierbarkeitsuntersuchungen vor dem Produktiveinsatz.

Danksagung

Diese Arbeit wird durch das Bundesministerium für Bildung und Forschung (BMBF) im Rahmen der D-Grid Initiative (F&L-Grid) finanziell unterstützt (`http://www.bmbf.de/`).

Literatur

[ACHH+07] M. Antonioletti, N.P. Chue Hong, A.C. Hume, M. Jackson, K. Karasavvas, A. Krause, J.M. Schopf, M.P. Atkinson, B. Dobrzelecki, M. Illingworth, N. McDonnell, M. Parsons, and E. Theocharopoulos. OGSA-DAI 3.0 The Whats and the Whys. In *Proceedings of the UK e-Science All Hands Meeting*, 2007.

[BMB] German Federal Ministry of Education and Research (BMBF). `http://www.bmbf.de/`.

[DFN] German Research Network (DFN). `http://www.dfn.de/`.

[FKT01] I. Foster, C. Kesselman, and S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of High Performance Computing Applications*, 15:200–222, 2001.

[JSc] JSch – Java Secure Channel (Project Homepage). `http://www.jcraft.com/jsch/`.

[Mic] Sun Microsystems. Java 2 Platform, Standard Edition. `http://java.sun.com/j2se/1.4.2/download.html`.

[NKG07] H. Neuroth, M. Kerzel, and W. Gentzsch. *German Grid Initiative (D-Grid)*. Niedersächsische Staats- und Universitätsbibliothek, 2007.

[OAS] OASIS. Web Services Resource Framework (WSRF). `http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrf`.

[Pet08] D. Petcu. A Comprehensive Development Guide for the Globus Toolkit. *Distributed Systems Online*, 9:4–6, 2008.

[Rom99] M. Romberg. The UNICORE Architecture: Seamless Access to Distributed Resources. In *Proceedings of the* 8^{th} *IEEE International Symposium on High Performance Distributed Computing (HPDC)*, pages 287–293. IEEE Computer Society Press, 1999.

[SJWA06] R.O. Sinnott, J. Jiang, J. Watt, and O. Ajayi. Shibboleth-based Access to and Usage of Grid Resources. In *Proceedings of the* 7^{th} *IEEE/ACM International Conference on Grid Computing*, pages 136–143. IEEE Computer Society Press, 2006.

[W3C06] W3C. Web Services Description Language (WSDL) 2.0, June 2006. `http://www.w3.org/TR/wsdl20/`.

[ZKI] Zentren für Kommunikation und Informationsverarbeitung in Lehre und Forschung e.V. (ZKI), Arbeitskreis Netzdienste. `http://www.zki.de/ak_nd/`.

Netztechnologien

Konzept und Design einer autonom funktionsfähigen Knoten-Plattform für Wireless Mesh Backbones

Alexander Gladisch, Martin Arndt, Robil Daher, Martin Krohn, Djamshid Tavangarian

Lehrstuhl für Rechnerarchitektur
Universität Rostock
Albert-Einstein-Straße 21
D-18059 Rostock
vorname.nachname@uni-rostock.de

Abstract: Wireless Mesh Netzworks (WMNs) werden aufgrund ihrer Flexibilität als effiziente, drahtlose Alternative zur Versorgung von strukturschwachen Gebieten eingesetzt. Das Design vollständig autonom funktionsfähiger Mesh-Knoten stellt jedoch noch immer eine große Herausforderung dar. Basierend auf einer modularen Netzwerkarchitektur wird in diesem Artikel das Konzept für eine Commercial Off-The-Shelf (COTS) Hardware und Open-Source-Software (OSS) basierte Mesh-Knoten-Plattform entwickelt, welche den unabhängigen Betrieb eines solchen Kommunikationsnetzwerkes ermöglicht. Dabei werden auch Konzepte zur Energieversorgung berücksichtigt. Resultierend aus dieser Entwicklung werden beispielhaft zwei funktionsfähige Prototypen präsentiert.

1 Einführung

Wireless Mesh Netzworks (WMNs) werden aufgrund ihrer Flexibilität als effiziente, drahtlose Alternative zur Versorgung strukturschwacher Gebiete eingesetzt. Dies trifft nicht nur auf kleinere Städte oder ländliche Gemeinden zu, in denen kein drahtgebundener Breitbandanschluss verfügbar ist [BW08], sondern auch international, z.B. in Entwicklungsländern oder abgelegenen Regionen ohne Infrastruktur. WMNs sind besonders flexibel, da sie in der Regel aus autonomen (d.h. unhabhängig von der vorhandenen Infrastruktur und ohne zentrale Netzwerkarchitektur funktionsfähigen) Knoten bestehen, die miteinander kooperieren, um Clients mit den benötigten Netzwerkdiensten zu versorgen [AW05]. Zurzeit existieren verschiedene Netzwerktechnologien, die Mesh-Eigenschaften unterstützen, darunter sind u. a. UMTS-LTE, IEEE 802.16 WiMAX, IEEE 802.11 WLAN und IEEE 802.15 WPAN. Theoretisch können diese Technologien sowohl in Backbone- als auch in Zugriffs-Netzwerken verwendet werden, praktisch stellt IEEE 802.11 in WMNs momentan für beide Netzwerkschichten die meistverwendete Technologie dar [LZ06]. Aufgrund der weiten Verbreitung der Technologie ist WLAN besonders kostengünstig, weshalb sie bei der Entwicklung einer autonomen Knoten-Plattform vordergründig berücksichtigt wird.

Das Aufbauen autonomer Mesh-Knoten stellt eine große Herausforderung dar, da deren Selbstständigkeit auf zwei Ebenen betrachtet werden muss: Kommunikationsnetzwerk und Energieversorgung. Im Kommunikationsnetzwerk sind Techniken wie Selbst-Organisation und -Konfiguration sowie Selbst-Heilung zu unterstützen. Dafür müssen

verschiedene Routing-Protokolle, QoS-Mechanismen, etc. berücksichtigt werden. Für eine autonome Energieversorgung müssen verschiedene alternative Energiequellen in Kombination mit Energiespeichern wie Akkumulatoren analysiert werden. Der Einsatz eines geeigneten Energie-Management-Mechanismus ist dabei obligatorisch.

Der aktuelle Stand der Technik zeigt, dass die Autonomie der Knoten durch aktuell verfügbare Plattformen bisher nicht oder nur unzureichend gewährleistet ist. In dieser Arbeit werden die Anforderungen an eine solche Plattform sowohl für das Kommunikationsnetzwerk als auch für die Energieversorgung beschrieben und ein Konzept für ein entsprechendes Plattformdesign auf Basis von Commercial Off-The-Shelf (COTS) Hardware und Open-Source Software (OSS) erstellt. Anschließend werden zwei Prototypen präsentiert, die basierend auf den Ergebnissen des vorgestellten Konzeptes aufgebaut wurden.

Im Weiteren ist diese Arbeit wie folgt gegliedert: In Kapitel 2 wird der aktuelle Stand der Technik beschrieben. Kapitel 3 stellt beispielhaft die Architektur eines WMNs vor, in dem das entwickelte Plattformdesign eingesetzt werden kann. Kapitel 4 beschreibt das Konzept für den Aufbau einer Knoten-Plattform. Dabei wird zwischen Netzwerkkomponenten, Protokollen und Diensten sowie Hard- und Softwareplattform unterschieden. Nachfolgend werden in Kapitel 5 zwei konkrete Prototypen vorgestellt. Schließlich wird der Artikel in Kapitel 6 zusammengefasst.

2 Stand der Technik

In Aufbau und Implementierung unterscheiden sich die verfügbaren Lösungen von 802.11-basierten Knoten-Plattformen in zwei Design-Philosophien [Bu03]: Hardware-Defined Radio (HDR) und Software-Defined Radio (SDR).

In der HDR-Technologie ist die Hardware-Plattform für die Funkschnittstelle komplett verantwortlich, während die Software-Plattform, in Form von Betriebssystem und Treibern, der Anbindung an Nutzerapplikationen sowie Netzwerkservices dient. Unternehmen wie Cisco, Lucent-Alcatel, 3COM und andere stellen ihre eigenen Plattform-Produkte für Hardware und Software her. Für ihren angestammten Anwendungszweck bieten derartige Produkte ausreichend Leistung, stellen jedoch proprietäre Lösungen dar. Die Mehrheit dieser ist nur von den Herstellern selbst mit vielen Beschränkungen in Hard- und Software erweiterbar. Auch wenn einige Produkte mit OSS-Plattformen angeboten werden, wie z.B. WLAN-AP/Router von LinkSys und Asus, bleiben die Beschränkungen von Hardware-Ressourcen ein wesentliches Problem für die Integration neuer Software- und Hardware-Module, was sich z. B. in einer Speicher-Beschränkung für ein neues, speicherintensiveres Routing-Protokoll, oder einer mangelnden Erweiterbarkeit um eine neue WLAN-Karte zeigt. Ganz im Gegensatz zu diesen fertigen Produkten werden skalierbarere Lösungen meist basierend auf COTS Hardware-Komponenten und OSS aufgebaut [RM08]. Die entsprechenden Hardware-Plattformen sind meist mit General-Purpose-CPUs (ARM, XScale, x86, etc.) ausgerüstet. Als Bauform der Mainboards werden oft PC/104, ECX oder nicht standardisierte Größen, ausgestattet mit mehreren mini-PCI-Slots zur Anbindung von WLAN-Karten, eingesetzt. Hardwarekomponenten wie

Netzwerkschnittstellen, Antennen, etc. werden in Zusammenhang mit System-Design und Netzwerkplanung eingebaut. Ein Beispiel für eine Firma, die mit vorgefertigten Mesh-Knoten dieses Design verfolgt ist Saxnet [Sa09]. Die Software-Plattform ist meist Linux basiert, wobei spezielle Linux-Distributionen wie DSL, Puppy oder das stark modifizierbare Buildroot mit kleinem Footprint verwendet werden. Zusätzliche Software-Komponenten werden nach Bedarf integriert, insbesondere Module für Routing-Protokolle, Netzwerk-Services, Nutzer-Applikationen, etc. In der Summe entstehen komplette Knoten-Plattformen.

In der SDR-Technologie hingegen wird die Funkschnittstelle nur partiell, d. h. nur der Signal-Empfang und -Versand, in Hardware implementiert, während die gesamte Signalbearbeitung von der Software-Plattform durchgeführt wird. Nach unsererem Kenntnisstand bietet das GNU Radio Open-Source-Projekt, als Software-Plattform, in Zusammenhang mit der ETTUS USRP und einer General-Purpose-Prozessor-basierenden Hardware-Plattform, momentan die einzige SDR-technologie-basierende Open-Source-Lösung für WLAN-Knoten, wobei zusätzliche OSS-Pakete für den Aufbau einer vollständigen Knoten-Plattform benötigt werden. Dickens et. al [DD08] beschreibt den Aufbau eines solchen Knotens für Forschungszwecke, wobei der vorgestellte Prototyp beschränkte Erweiterungsmöglichkeiten, insbesondere der Funkschnittstellen, anbietet.

Beide beschriebenen Möglichkeiten, sowohl die verfügbaren HDR-basierten Lösungen, als auch die verfügbaren Lösungen von SDR-Technologie decken die Anforderung an Design- und System-Flexibilität in Zusammenhang mit der erwarteten Funktionalität und hoher Erweiterbarkeit sowie Skalierbarkeit von Netzwerk-Knoten kaum ab. Alternativ bietet die Kombination von OSS und COTS-Hardware-Komponenten eine vielversprechende Variante zum Aufbau flexibler Knoten-Plattformen, welche die entsprechenden Anforderungen erfüllen sollte.

3 Beispiel einer modularen Netzwerkarchitektur

Als Beispiel für den Einsatz der Knoten-Plattform soll eine modulare Multi-Layer Architektur dienen, die für die Versorgung von infrastrukturschwachen Gebieten vorgesehen ist. In ihr sind drei grundlegende Netzwerk-Ebenen definiert: Zugriffsnetz (Access), Backbone und Zubringer (Supply/Internet), wie in Abbildung 1 dargestellt. Im Netzwerk können drei Haupttypen von Knoten nach dem IEEE 802.11s-Draft unterschieden werden: (1) Der Mesh Point (MP) als Relay-Station (RS) dient zur Paket-Weiterleitung in der Backbone-Ebene; (2) Der Mesh Access Point (MAP) ermöglicht zum einen den Zugriffs der Clients (Mobile Station – MS) auf das Netzwerk und zum anderen die Anbindung an die Backbone-Ebene; (3) Das Mesh Point Portal (MPP) dient als Gateway-Station (GS) zur Anbindung der Backbone-Ebene an die Zubringer-Ebene. Zur Überbrückung von großen Strecken und zur Erhöhung der Datenübertragungsrate können an allen Knoten der Ebenen spezialisierte Antennen (z.B. Yagi- oder omnidirektionale Antennen) verwendet werden. Die Anwendungsbereiche der beschriebenen Architektur reichen von der Versorgung von stationären Teilnehmern, wie Häusern oder Installationen, bis hin zur Unterstützung von mobilen Clients (Mobile Stations, MS), die sich innerhalb der abgedeckten Fläche bewegen.

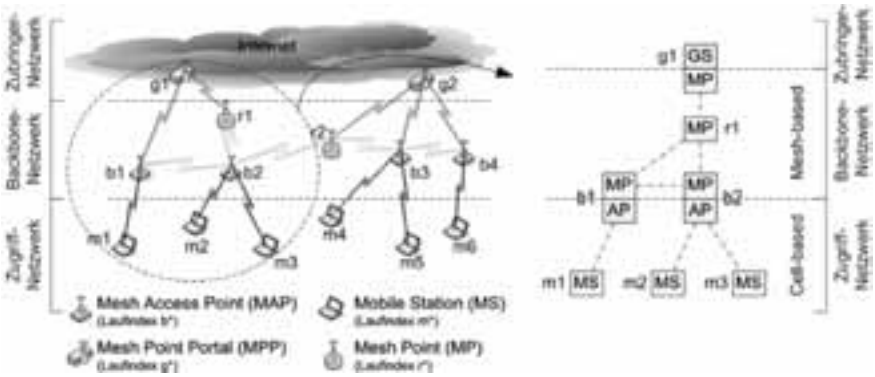


Abbildung 1: Darstellung der Beispielnetzwerkarchitektur

4 Konzept und Design der Knoten-Plattform

4.1 Netzwerkkomponenten

WLAN Technologien

Aufgrund der weiten Verbreitung des 802.11b/g Standards im Endkundenbereich, bietet sich dieser besonders zur Realisierung der Zugriffs-Ebene an. Durch die hohe Bandbreite, die 11 unabhängigen Kanäle im 5 GHz Band und die deutlich niedrigeren Interferenzen im 5 GHz Bereich, bietet der 802.11a/h Standard die beste Grundlage für den Einsatz in der Backbone-, bzw. Zubringer-Ebene [JP03]. Ebenfalls interessant für die Verwendung im Zubringer- und Backbone-Netzwerk ist der neue Draft 802.11n, sobald er endgültig als Standard verabschiedet wurde [DA03].

Schnittstellen und Antennen

Die Anzahl der drahtlosen Schnittstellen leitet sich aus dem Einsatzzweck und damit dem Knotentyp ab, der Sachverhalt ist in Tabelle 1 dargestellt. Die Einschränkung auf vier Schnittstellen ergibt sich aus der Anzahl zur Verfügung stehender Kanäle sowie der großen Anzahl der damit entstehenden Möglichkeiten zur Gestaltung der Netzwerktopologie. An dieser Stelle sollen zwei Fälle näher betrachtet werden: (1) Es ist nur ein Netzwerkinterface vorhanden, wobei für das gesamte Netzwerk nur ein gemeinsamer Kanal zur Kommunikation verwendet wird. Dabei werden omnidirektionale Antennen eingesetzt. In diesem Szenario sind der Datendurchsatz und die Resistenz des Systems auf Störeinflüsse relativ gering. Dagegen ist die Installation des Systems leicht durchzuführen, die Wartung wird vereinfacht und die Kosten des Gesamtsystems sind vergleichsweise niedrig. (2) Die Knoten besitzen jeweils vier Schnittstellen, wovon drei für den Backbone-Bereich eingesetzt und jede mit einem nicht überlappenden Kanal betrieben wird. Für die Verbindung der Knoten kommen gerichtete Antennen zum Einsatz. In diesem Szenario verfügt das gesamte Netzwerk über eine hohe Performance und ist resistent gegen Störeinflüsse. Es werden Algorithmen zur Kanalverteilung (z.B. Hyacinth [RC05]) benötigt, um Performance-Einbußen durch sich überlappende Kanäle zu vermeiden. Wartung und Installation des Systems sind komplex und erfordern einen erhöhten Aufwand.

Tabelle 1: Konfiguration der Knotentypen

Knotentyp	Anzahl (A) der drahtlosen Netzwerkschnittstellen	Eingesetzte Technologien	Eingesetzte Antennentypen
MP	$1 \leq A \leq 4$	802.11a, 802.11n (5 GHz)	gerichtet
MAP	$2 \leq A \leq 4$	802.11a, 802.11n (2,4 bzw. 5 GHz), 802.11g	gerichtet, omnidirektional
MPP	$1 \leq A \leq 4$	802.11a, 802.11n (5 GHz)	gerichtet

4.2 Protokolle und Dienste

Autonomie und Fernwartung
Um die Selbstorganisation und –konfiguration der Knoten zu gewährleisten, muss eine möglichst modulare Software-Plattform entwickelt werden (eine entsprechende Eigenentwicklung wird in Kapitel 4.3 beschrieben), in die alle für den optimalen Betrieb eines Netzwerks notwendigen Funktionen integriert werden können. So sind ein Mechanismus zur strukturierten Vergabe oder zur dynamischen Konfiguration von IP-Adressen (z.B. über DHCP oder eigens entwickelte Adressvergabemechanismen), ein Algorithmus zur Verteilung von Funk-Kanälen der drahtlosen Schnittstellen (z.B. Hyacinth [RC05]) sowie eine geeignete Power-Management-Software (die aktuell von den Autoren entwickelt wird) für den dauerhaften und autonomen Betrieb der Plattform unverzichtbar. Um die Plattform effizient wartbar zu gestalten, sind Geräte- bzw. Funktionsparameter fernadministrierbar (z.B. über SNMP oder SSH) zu entwerfen.

Routing
Für die Wahl geeigneter Kommunikationspfade innerhalb des Netzwerks muss ein geeignetes Protokoll im Software-Framework bereitstehen. Insbesondere für drahtlose Netzwerke aller Art (Ad-hoc Netzwerke, MANETs und WMNs) steht eine große Anzahl verschiedener Routing-Protokolle zur Verfügung. In [HX02] werden die verschiedenen Design-Philosophien von Routing-Protokollen (proaktiv, reaktiv, hierarchisch, geographisch) sowie typische Vertreter (OLSR, AODV, uvm.) beschrieben und klassifiziert. Die Leistungsfähigkeit des eingesetzten Routing-Protokolls hängt stark von der im Protokoll genutzten Metrik ab. Neben den in den Protokollen genutzten Standard-Metriken können auch Metriken wie Expected-Transmission-Count (ETX) [CA03] bzw. -Time (ETT) und deren Erweiterung [DP04] in drahtlose Routing-Protokolle integriert werden. Als besonders effizient für Mesh-Netzwerke hat sich die Metrik EMO (Extended Medium Observation, [Ko08] erwiesen. Welches Protokoll sich für den Einsatz auf der Knoten-Plattform eignet, hängt also stark vom Einsatzszenario ab und muss nach Bedarf entschieden werden. Gegebenenfalls ist für spezielle Anwendungsfälle die Entwicklung eigener Lösungen und deren Integration in das Framework sinnvoll.

Quality of Service und Lastbalancierung
Um die vorhandenen Netzwerk-Ressourcen optimal nutzen zu können, werden geeignete Mechanismen zur Verteilung der im Netzwerk entstehenden Last benötigt, die in das Software-Framework eingebunden werden können. Durch Lastbalancierung kann es dem Gesamtsystem ermöglicht werden, Ressourcen- sowie Mobilitätsmanagement zu unterstützen und dadurch QoS-Eigenschaften für die Anbindung der Teilnehmer zu

verbessern. Dies kann beispielsweise mittels QoS-orienterter Lastbalancierungs-Mechanismen wie in [DT06] erreicht werden. Weitere Möglichkeiten stellen IntServ (RFC2210) oder DiffServ (RFC2475) sowie die Unterstützung des Standards 802.11e dar. Auch Eigenentwicklungen können bei geeigneter Integration in das Software-Framework verwendet werden.

Sicherheit
Grade in frei zugänglichen drahtlosen Netzwerken spielen Sicherheitsmechanismen eine wichtige Rolle. Insbesondere die Mechanismen der Protokollerweiterung 802.11i, welche die Authentifizierungs- und Autorisierungsmechanismen des 802.1X Standards enthält und die AES-Verschlüsselung unterstützt, müssen in der Knoten-Plattform implementiert werden. Als Sicherheitsprotokolle können EAP-TLS oder EAP-TTLS in Zusammenhang mit RADIUS zum Einsatz kommen. Ebenso sollte eine passende Software zur Systemüberwachung und zum Systemmanagement integriert werden.

4.3 Software-Plattform

Modularität
Um für die genutzten Protokolle und Dienste eine einheitliche Schnittstelle auf die benötigten Funktionen des Betriebssystems wie z.B. den Netzwerk-Stack oder die Treiber zu schaffen und eine einheitliche Datenbasis für alle Protokolle und Dienste vorzuhalten, bietet sich eine Agenten-basierte Softwarearchitektur an. Das Grundgerüst des Agenten stellt die genannte Funktionalität über definierte Schnittstellen bereit. Die z.B. als Plugin realisierten Protokolle und Dienste können über diese Schnittstellen die von ihnen benötigten Daten abrufen und entsprechende Steuerbefehle auslösen. Somit wird weiter vom Betriebssystem und den verwendeten Treibern abstrahiert und eine modular erweiterbare Firmware geschaffen.

Betriebssystem
Um die erforderliche Grundfunktionalität für die benötigten Dienste bereitzustellen, wird ein geeignetes Betriebssystem (BS) benötigt. Dieses sollte möglichst diverse Hardwarekomponenten und Prozessor-Architekturen unterstützen, damit die Software-Plattform flexibel einsetzbar bzw. portabel ist. Besonders gut für diesen Einsatzzweck geeignet ist Linux. Durch den frei verfügbaren Quellcode ist es flexibel anpassbar, auch der Großteil der für den Betrieb der Plattform benötigten Funktionen ist in Linux bereits enthalten. Die Entwicklung eigener Software für Linux-Systeme ist problemlos möglich, da eine Vielzahl von Programmiersprachen mit zugehörigen Compilern für die unterschiedlichsten Hardware-Plattformen zur Verfügung stehen. Um ein möglichst Ressourcen sparendes System zu erstellen, sollte ein speziell auf die Hardware-Plattform angepasstes, auf Linux basierendes Buildroot-System [Br09] verwendet werden. Es beinhaltet nur die genau für diesen Einsatzzweck benötigten Softwarekomponenten, was die optimale Nutzung der vorhandenen Ressourcen sicherstellt.

Treiber
Um die Funktionalität der eingesetzten Protokolle und Dienste bereitzustellen, müssen die Treiber der drahtlosen Netzwerkschnittstellen besondere Anforderungen erfüllen. Der Treiber muss in der Lage sein, detaillierte Informationen über das drahtlose Medium zu liefern und Befehle für die Steuerung der Hardware (außerhalb der normalen

Funktionen im End-Kunden-Bereich) entgegenzunehmen. Für Linux bieten sich daher insbesondere die Treiber Madwifi [Ma09] bzw. die Nachfolger Ath5k und Ath9k [Wk09] für WLAN-Karten mit Atheros-Chipsätzen an, da sie diese Anforderungen erfüllen. Um den Funktionsumfang der Treiber noch zu erweitern, können sie mit dem Linux-Programm HostAP kombiniert werden.

4.4. Hardware-Plattform

Jede Hardware-Plattform besitzt einige grundlegende Komponenten. Sie besteht aus einem Mikrocomputer mit einem zentralen Prozessor, Arbeitsspeicher, Festspeicher und verschiedenen Ein- und Ausgabeschnittstellen. Dazu gehören Netzwerkschnittstellen, serielle Schnittstellen zur Behebung von Störungen sowie USB- und Mini-PCI-Schnittstellen zur einfachen Erweiterung sowie zur Erhöhung der Flexibilität des Geräts, z. B. mit zusätzlichem externen Speicher. Um die Plattform so wartungsarm wie möglich zu realisieren, muss auf bewegliche Teile wie Festplatten und Lüfter verzichtet werden, da diese Komponenten besonders fehleranfällig sind [GH99]. Um die Komponenten der Plattform vor Umwelteinflüssen zu schützen, wird weiterhin ein wetterfestes Gehäuse benötigt. Im Folgenden wird auf die einzelnen Kriterien detaillierter eingegangen.

Mikroprozessor
Als Basis des Systems stehen verschiedene Mikroprozessor-Architekturen zur Auswahl. Neben der weit verbreiteten x86-Architektur sind für eingebettete Systeme insbesondere die ARM-Architektur [Ar09] und entsprechende Derivate davon (z.B. XSCALE von Intel) interessant. Aktuelle ARM-Prozessoren haben im Gegensatz zu x86-Prozessoren einen besonders geringen Stromverbrauch, was die autonome Stromversorgung der Plattform enorm erleichtert. Zusätzlich wird wenig Abwärme erzeugt, was eine lüfterlose und somit wartungsärmere Plattform ermöglicht. Da für diese Architektur entsprechende Compiler für Linux existieren, ist ihr Einsatz in der Plattform empfehlenswert.

Netzwerkschnittstellen
Damit die Hardware-Plattform universell eingesetzt werden kann, müssen die über Mini-PCI oder USB angeschlossenen drahtlosen Netzwerkschnittstellen alle erforderlichen WLAN-Standards unterstützen. Dazu gehören die Standards 802.11a/b/g sowie der Draft 802.11n zum Betrieb des Physical- und des MAC-Layers, aber auch weitere Funktionalität wie Sicherheit, die mit der Standarderweiterung 802.11i bereitgestellt wird. Zusätzlich muss die Netzwerkschnittstelle mit den in der Software-Plattform beschriebenen Treibern betrieben werden können. Dies trifft insbesondere auf WLAN-Karten mit aktuellen Atheros-Chipsätzen zu, weshalb deren Einsatz in der Hardware-Plattform empfohlen wird. Zur Integration der Netzwerkkarten sollte Mini-PCI bevorzugt werden, da es höhere Datenraten als USB zur Verfügung stellt. USB 2.0 (mit Datenraten bis zu 480 MBit/s) stößt bei Anschluss von Draft 802.11n Hardware (mit Datenraten bis zu 600 MBit/s) bereits an seine Leistungsgrenzen.

Stromversorgung
Für den autonomen Betrieb der Plattform wird eine adäquate Stromversorgung benötigt. Hier bietet sich der Einsatz (verschiedener) alternativer Energiequellen wie z.B. Solarzellen oder Windkraft in Kombination mit Energiespeichern wie Akkumulatoren an. Dies ermöglicht die eigenständige Erzeugung der von der Plattform benötigten

Energie und befähigt den Knoten auch Zeiträume, in denen die Energieerzeugung nicht möglich ist, zu überbrücken. Die Steuerung der Energieversorgung übernimmt eine spezielle Power-Management-Software. Je nach geographischer Lage der einzelnen Netzwerkknoten können zudem mehrere Knoten des WMNs zu einem Verbund zusammengefasst und über eine einzelne, besonders leistungsfähige Energiequelle betrieben werden. Der Anschluss der Energiequelle an die Plattform erfolgt über Power-over-Ethernet (PoE) oder einen Niederspannungs-Gleichstrom-Anschluss.

Realisierungsformen
Eine Plattform mit den geforderten Eigenschaften kann, wie in [RM08] beschrieben als Personal Computer (PC) oder Eingebettetes System realisiert werden. Dabei erscheint die Realisierung als Eingebettetes System besonders vorteilhaft. Es basiert auf einer großen Auswahl gut verfügbarer und somit kostengünstiger Hardwarekomponenten und ist bei ausreichender Leistung zum Betrieb der Plattform besonders energiesparend.

5 Prototypische Umsetzung

Abbildung 2: Prototyp 1 (indoor)

Abbildung 3: Prototyp 2 (outdoor)

Anhand der gegebenen Empfehlungen und des erstellen Konzepts der Knotenplattform, wurden zwei Prototypen konstruiert. Um die Plattform im Access- und im Backbone-Bereich von Netzwerken einsetzen zu können, wurde vorrangig auf eine möglichst flexible Zusammenstellung der einzelnen Komponenten geachtet. Es wurde bewusst das genannte Maximum von vier drahtlosen Netzwerkschnittstellen integriert, um eine Vielzahl von Anwendungsszenarien abzudecken. So können die beiden entstandenen prototypischen Knoten-Plattformen zur Implementierung von MP, AP, MAP oder auch MPP mit vier drahtlosen und einer zusätzlichen drahtgebundenen Netzwerkschnittstelle dienen. Beide Prototypen haben gemein, dass sie momentan ohne Power-Management über das Stromnetz bzw. Akku betrieben werden. Die weiterentwickelte Version des zweiten Prototyps wird auf Basis von Solar-Energie in Kombination mit Akkus betrieben, wobei eine Power-Management-Software eingesetzt wird.

5.1 Prototyp für den Indoor-Bereich

Um die generelle Umsetzbarkeit zu verifizieren, wurde diese Knoten-Plattform zunächst als Indoor-Variante realisiert. Die Plattform besteht aus den in Tabelle 2 genannten Hardware-Komponenten und wird mit einer speziell zusammengestellten Software-Plattform betrieben, die alle in Kapitel 4 definierten Bestandteile enthält. Eine Darstellung des ersten Prototyps ist in Abbildung 2 zu sehen.

Tabelle 2: Komponenten der Prototypen

Komponente		Prototyp 1 (Indoor-Bereich)	Prototyp 2 (Outdoor-Bereich)
Protokolle u. Dienste	Adressvergabe	DHCP	DHCP
	Routing-Protokoll	OLSR	AODV basiert auf EMO
	Quality of Service	Nicht integriert	Nur 802.11e
	Sicherheitsmechanismus	WPA	WPA2 + RADIUS
Softwareplattform	Betriebsystem	Buildroot/Linux: Eigene Zusammenstellung	Buildroot/Linux: Eigene Zusammenstellung
	WLAN-Treiber	Madwifi	Atheros 5k
	Root-Dateisystem	Modifiziertes Buildroot Remote-Update möglich	Modifiziertes Buildroot Remote-Update möglich
Hardwareplattform	Prozessor	Intel Celeron 400 MHz	Intel XScale IXP 425 533 MHz
	Hauptplatine	PC104-Board mit 4 Mini-PCI-Steckplätzen, 2 Ethernet-Schnittstellen 2 RS232-Schnittstellen	Gateworks Avila GW2348 mit 4 Mini-PCI-Steckplätzen, 2 Ethernet-Schnittstellen, 2 RS232 Schnittstellen
	Festwertspeicher	1 GB CF-Karte	512 MB CF-Karte
	Funknetzwerk-Schnittstellen	4 WLAN 802.11b/g Mini-PCI Steckkarten (Atheros-Chipsatz)	4 WLAN 802.11a/b/g Mini-PCI Steckkarten (Atheros-Chipsatz)
	Hauptspeicher	128 MB SDRAM	64 MB SDRAM
	Gehäuse	Eigenentwicklung	Gateworks GW3020, witterungsfest

5.2 Prototyp für den Outdoor-Bereich

Dieser Prototyp (Abbildung 3) wurde im Vergleich zum ersten hinsichtlich der Erfüllung der Anforderungen deutlich verbessert. So ist sein Gehäuse nun auch für den Einsatz im Outdoor-Bereich geeignet, die Energieversorgung wurde auf PoE umgestellt. Durch die Erweiterung der Funk-Netzwerkschnittstellen von 802.11b/g auf 802.11a/b/g Technologie, wird es dem Backbone-Netzwerk ermöglicht, im 5 GHz Frequenzband zu arbeiten und so Interferenzen zu minimieren bzw. die Leistungsfähigkeit zu optimieren. Ebenfalls wurde die Gerätegröße reduziert, unnötige Bestandteile wurden entfernt. Der Prototyp besteht aus den in Tabelle 2 genannten Komponenten und verwendet eine weiterentwickelte Version der Software-Plattform der ersten Version.

6 Zusammenfassung und Ausblick

Im vorliegenden Artikel wurden das Konzept eines autonom funktionsfähigen Mesh-Knotens vorgestellt. Auf Basis der beschriebenen Anforderungen an einen solchen Mesh-Knoten wurde eine entsprechende Knoten-Plattform entwickelt, wobei das Netzwerkdesign sowie die Hard- und Software-Architektur differenziert betrachtet wurden. Während beim Netzwerkdesign insbesondere auf die Konfiguration der Netzwerkkomponenten und verwendete Technologien eingegangen wurde, wurde beim Design der Softwareplattform die zugrunde liegende Open-Source Systemsoftware sowie die benötigten Protokolle und Dienste beschrieben. Beim Design der Hardwareplattform wurden verschiedene Architekturen der Basis-Plattform betrachtet

und ihr autonomer Betrieb sowie mögliche Realisierungsformen mit COTS-Hardware fokussiert. Weiterhin wurden als Beispielimplementierung zwei funktionsfähige Prototypen vorgestellt. Um die Leistungsfähigkeit der Prototypen weiter zu steigern, findet aktuell eine weitere Optimierung der Hard- und Software-Plattform statt.

Literaturverzeichnis

[Ar09] Webseite von ARM, http://www.arm.com

[AW05] Akyildiz, I. F.; Wang, X.; Wang, W.: Wireless Mesh Networks: A survey. In: Computer Networks Journal (Elsevier), 2005.

[Ba06] Bundesnetzagentur: Allgemeinzuteilung von Frequenzen in den Bereichen 5150 MHz - 5350 MHz und 5470 MHz - 5725 MHz für Funkanwendungen zur breitbandigen Datenübertragung, WAS/WLAN (Wireless Access Systems including Wireless Local Area Networks), 2006

[Br09] Buildroot Webseite, http://buildroot.uclibc.org/

[Bu03] Burns, P.G.: Software Defined Radio for 3G. Published by Artech House, 2003

[Bw08] Bundesministerium für Wirtschaft und Technologie: „Breitband-Verfügbarkeit in Deutschland“, http://www.zukunft-breitband.de/BBA/Navigation/Breitbandatlas/laenderkarten.html?, 2008

[CA03] De Couto, D., Aguayo, D., Bicket, J., Morris, R.: A High-Throughput Path Metric for Multi-Hop Wireless Routing, ACM Mobicom 2003

[CB03] Chapin, J.M.; Bose, V.G.: Vanu Software Radio System, Vanu Inc., www.vanu.com, 2003

[DA03] Doefexi, A.; Armour, S.; Beng-Sin Lee; Nix, A.; Bull, D.: An evaluation of the performance of IEEE 802.11a and 802.11g wireless local area networks in a corporate office environment. IEEE International Conference on Communications, 2003

[DD08] Dickens, M.L.; Dunn, B.P.; Laneman, J.N.: Design and Implementation of a Portable Software Radio, IEEE Communication Magazine, August 2008

[DP04] Draves, R., Padhye, J., Zill, B.: Routing in Multi-Radio, Multi-Hop Wireless Mesh Networks, ACM Mobicom 2004

[DT06] Daher, Robil; Tavangarian, Djamshid: “QoS-oriented Load Balancing for WLANs”, In Proc. of The First International Workshop on Operator-assisted (Wireless Mesh) Community Networks 2006 (OpComm'06), Berlin, Germany, September 18-19, 2006

[GH99] Gebauer, J.; Hartmann, H.; Seguin, M.: Clustering mit Windows NT. 1. Auflage, Addison-Wesley, Bonn 1999

[HX02] Hong, X., Xu, K., Gerla, M.. Scalable routing protocols for mobile ad hoc networks. Network, IEEE , vol.16, no.4, pp.11-21, Jul/Aug 2002

[JP03] Jain, K., Padhye, J., Padmanabhan, V. N., Qiu, L.: Impact of interference on multi-hop wireless network performance. In: ACM Annual International Conference on Mobile Computing and Networking (MOBICOM), 2003

[Ko08] Kopp, Heiko; Krohn, Martin; Tavangarian, Djamshid,: Extended Medium Observation (EMO) — A load-aware metric for routing in wireless mesh backbone networks, Software, Telecommunications and Computer Networks, 2008. SoftCOM 2008. 16th International Conference on , 2008

[LZ06] Lee, M. J.; Zheng, M.; Ko, J. B.; Shrestha, D.M.: Emerging standards for wireless mesh technolog, IEEE Wireless Communications, vol. 13, no. 2, pp. 56 - 63, April 2006

[Ma09] Webseite des Projekts Madwifi, http://madwifi-project.org/

[RC05] Raniwala, A.; Chiueh, T.: Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network. In Proc. of IEEE INFOCOM, 2005.

[RM08] Riggio, R.; Miorandi, D.; Chlamtac, I.; Scalabrino, N.; Gregori, E.; Granelli, F.; Yuguang F.: Hardware and software solutions for wireless mesh network testbeds, IEEE Communications Magazine, vol.46, no.6, pp.156-162, June 2008

[Sa09] Webseite der Firma Saxnet, http://saxnet.de/

[Sj03] Steinheider, J.: Software-Defined Radio Comes of Age, Vanu Inc., www.vanu.com, 2003

[Wk09] Webseite des Linux Wireless Wikis, http://wireless.kernel.org/

MPLS-TP – The New Technology for Packet Transport Networks

Dieter Beller, Rolf Sperber

FS/O/PDL, FS/R/VP
Alcatel-Lucent Deutschland AG
Lorenzstraße 10
D-70435 Stuttgart
Dieter.Beller@alcatel-lucent.com
Rolf.Sperber@alcatel-lucent.de

Abstract: The Internet Engineering Task Force (IETF) and the Telecommunication Standardization Sector of the International Telecommunication Union (ITU-T) have undertaken a joint effort to standardize a new transport profile for the multi-protocol label switching (MPLS) technology that is intended to provide the basis for the next generation packet transport network. The fundamental idea of this activity is to extend MPLS where necessary with Operations, Administration and Maintenance (OAM) tools that are widely applied in existing transport network technologies such as SONET/SDH or OTN. This paper provides a brief history of the MPLS-TP standardization activities and addresses the MPLS-TP OAM functions. These functions are targeted at making MPLS comparable to SONET/SDH and OTN in terms of reliability and monitoring capabilities, i.e., MPLS-TP will become a true carrier grade packet transport technology. An MPLS-TP network can be operated in an SDH-like fashion and a network management system (NMS) can be used to configure connections. Connection management and restoration functions, however, can alternatively be provided utilizing the Generalized MPLS (GMPLS) control plane protocols which are also applicable to the MPLS-TP data plane. In addition to the simplification of the network operation leading to reduced operational expenditures (OPEX), the GMPLS control plane provides network restoration capabilities in addition to the network protection features that the MPLS-TP data plane already provides; this results in a further improved network resiliency. The MPLS-TP technology is also multi-service capable leveraging the pseudo-wire technology that has been developed at the IETF and which is still evolving. Some applications require synchronization, e.g. mobile services and interconnection of telephony switches. Ethernet is an asynchronous network protocol and hence protocol extensions are necessary. This paper discusses the different emerging standards. One of the key requirements is that the new MPLS-TP network layer must be capable to utilize the existing physical infrastructure and the paper lists the various adaptation or encapsulation techniques that allow MPLS-TP packets to be carried over a variety of different physical technologies ranging from SONET/SDH and OTN to Gigabit Ethernet.

1 Introduction

The purpose of a transport network is to provide a reliable aggregation and transport infrastructure for any client traffic type. With the growth of packet-based services, operators are transforming their network infrastructures while looking at reducing capital and operational expenditures. In this context, a new technology is emerging: a transport profile of Multi-Protocol Label Switching called MPLS-TP. MPLS-TP is currently under development at the IETF in collaboration with ITU-T experts. The objective of this standardization effort is to develop MPLS extensions where necessary in order to meet the transport network requirements depicted in Figure 1.

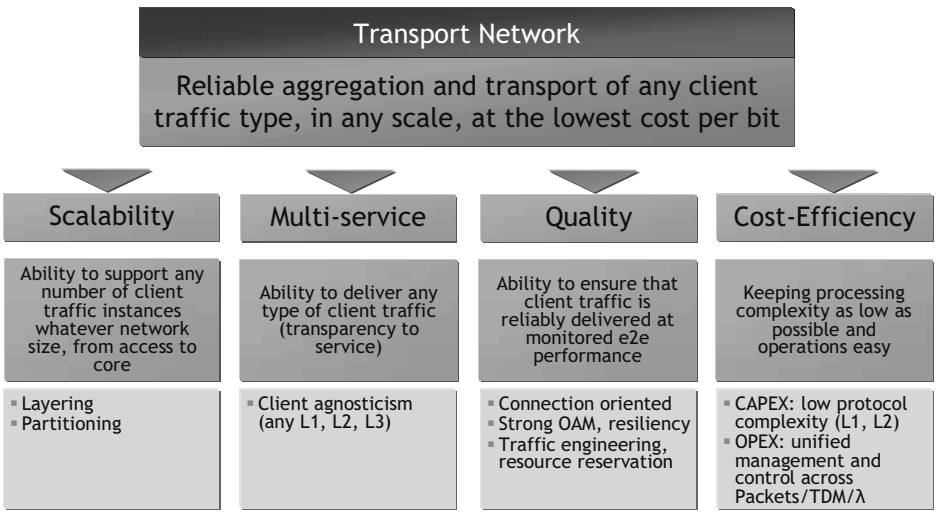


Figure 1: Transport Network Requirements

2 MPLS-TP Overview

The goal of MPLS-TP is to provide connection-oriented transport for packet and TDM services over optical networks leveraging the widely deployed MPLS technology. Key to this effort is the definition and implementation of OAM and resiliency features to ensure the capabilities needed for carrier-grade transport networks – scalable operations, high availability, performance monitoring and multi-domain support.

MPLS-TP key characteristics are:

- It is strictly connection oriented
- It is client-agnostic (can carry L3, L2, L1 services)
- It is physical layer agnostic (can run over IEEE Ethernet PHYs, SONET/SDH [G.783] and OTN [G.709],[G.872] using GFP [G.7041], WDM, etc.)

- It provides strong operations, administration and maintenance (OAM) functions similar to those available in traditional optical transport networks (e.g., SONET/SDH, OTN); these OAM functions are an integral part of the MPLS-TP data plane and are independent from the control plane
- It provides several protection schemes at the data plane similar to those available in traditional optical transport networks.
- It allows network provisioning via a centralized NMS and/or a distributed control plane
- The GMPLS control plane is also applicable to the MPLS-TP client or server layers and allows to use a common approach for management and control of multi-layer transport networks

Current transport networks (e.g. SONET/SDH) are typically operated from a network operation center (NOC) using a centralized network management system (NMS) that communicates with the network elements (NEs) in the field via the telecommunications management network (TMN, see ITU-T Recommendation M.3010 [M.3010]). The NMS provides well-known FCAPS management functions which are: fault, configuration, accounting, performance, and security management as defined in ITU-T Recommendation M.3400 [M.3400]. Together with survivability functions such as protection and restoration, availability figures of >99,999% have been achieved thanks to the highly sophisticated OAM functions that are existing e.g. in SONET/SDH transport networks. This well proven network management paradigm has been taken as basis for the development of the new MPLS-TP packet transport network technology.

Moreover, MPLS-TP provides dynamic provisioning of MPLS-TP transport paths via a control plane. The control plane is mainly used to provide restoration functions for improved network survivability in the presence of failures and it facilitates end-to-end path provisioning across network or operator domains. The operator has the choice to enable the control plane or to operate the network in a traditional way without control plane by means of an NMS. It shall be noted that the control plane does not make the NMS obsolete – the NMS needs to configure the control plane and also needs to interact with the control plane for connection management purposes.

2.1 Main Drivers for MPLS-TP

Carriers are experiencing an unprecedented combination of demand for service sophistication and expansion (e.g. Triple Play, LTE in mobile radio communications) coupled with economic pressure to minimize the cost for providing these services. MPLS-TP is being defined to meet these divergent requirements by introducing SDH-like OAM features to packet transport networks.

3 History of MPLS-TP Standardization

MPLS-TP started as Transport-MPLS at the ITU-T (see G.81xx series of ITU-T Recommendations), which was renamed to MPLS-TP based on the agreement that was reached between the ITU-T and the IETF to produce a converged set of standards for MPLS-TP.

3.1 T-MPLS Standardization Efforts at the ITU-T

Transport-MPLS (T-MPLS) was a standardization effort that was undertaken by the ITU-T. It is a packet-based transport network that will provide a key evolution path for next-generation networks reusing a profile of existing MPLS as defined by IETF and complementing it with transport-oriented OAM and protection capabilities. T-MPLS promises multi-service provisioning, multi-layer OAM and protection resulting in optimized circuit and packet resource utilization.

ITU-T approved the first version of its packet transport recommendation called Transport MPLS (T-MPLS) Architecture in 2006. By 2008, the technology had reached the stage where some vendors started supporting T-MPLS in their optical transport products. At the same time, the IETF was working on a new mechanism called Pseudo Wire Emulation Edge-to-Edge (PWE3) that emulates the essential attributes of a service such as ATM, TDM, Frame Relay or Ethernet over a Packet Switched Network (PSN), which can be an MPLS network [RFC3916].

A Joint Working Group (JWT) was formed between the IETF and the ITU-T to achieve mutual alignment of requirements and protocols.

3.2 MPLS-TP Standardization Efforts at the IETF

On the basis of the JWT activity, it was agreed that future standardization work will focus on defining MPLS-Transport Profile (MPLS-TP) within the IETF using the same functional requirements that drove the development of T-MPLS. When MPLS-TP RFCs will have reached a technical maturity level comparable with the existing T-MPLS Recommendations,, the ITU-T will align the latter with the MPLS-TP accomplishments from the IETF. The history and the process to produce a converged and consistent MPLS-TP standard consisting of IETF RFCs and ITU-T Recommendations is depicted in Figure 2.

Table 1 below provides an overview of the Internet Drafts on MPLS-TP that were published as of Mar 30, 2009 (Status: WG=working group draft, Ind.=individual draft). The MPLS-TP specifications are currently progressing at a good pace as the ITU-T G.81xx Recommendations already laid the foundations. The first stable IETF specifications for MPLS-TP are expected in 2009 and further expansions and refinements in 2010.

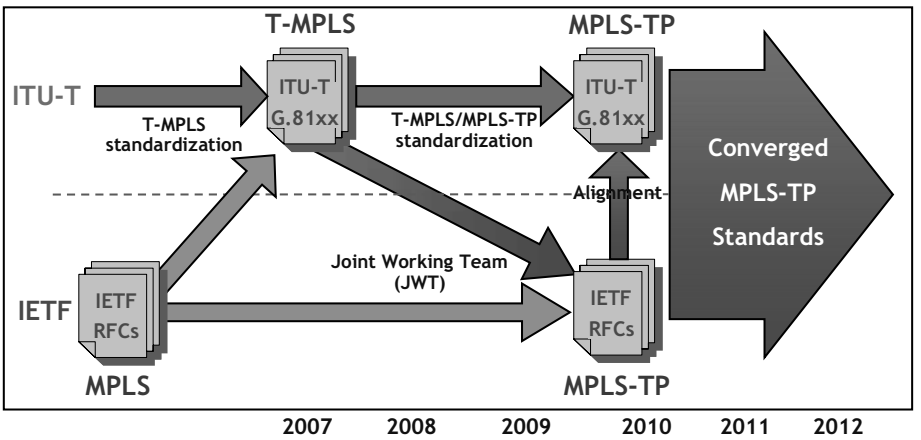


Figure 2: ITU-T/IETF Convergence towards Consistent MPLS-TP Standards

RFC / Internet-Draft	Title	Status
RFC 5317	JWT Report on MPLS Architectural Considerations for a Transport Profile	RFC
draft-ietf-mpls-tp-requirements-05	MPLS-TP Requirements	WG
draft-ietf-mpls-tp-framework-00	A Framework for MPLS in Transport Networks	WG
draft-ietf-mpls-tp-oam-requirements-01	Requirements for OAM in MPLS Transport Networks	WG
draft-ietf-mpls-tp-oam-framework-00	MPLS-TP OAM Framework and Overview	WG
draft-ietf-mpls-tp-nm-req-00	MPLS TP Network Management Requirements	WG
draft-ietf-mpls-tp-gach-gal-02	MPLS Generic Associated Channel	WG
draft-ietf-mpls-tp-gach-dcn-00	An Inband Data Communication Network For the MPLS Transport Profile	WG
draft-abfb-mpls-tp-control-plane-framework-00	MPLS-TP Control Plane Framework	Ind.
draft-andersson-mpls-tp-oam-def-01	"The OAM Acronym Soup"	Ind.
draft-andersson-mpls-tp-process-00	Joint IETF and ITU-T Multi-Protocol Label Switching (MPLS) Transport Profileprocess	Ind.
draft-bhh-mpls-tp-oam-y1731-01	MPLS-TP OAM based on Y.1731	Ind.
draft-boutros-mpls-tp-cv-01	Connection verification for MPLS Transport Profile LSP	Ind.
draft-boutros-mpls-tp-fault-01	Fault Management for the MPLS Transport Profile	Ind.
draft-boutros-mpls-tp-loopback-02	Operating MPLS Transport Profile LSP in Loopback Mode	Ind.
draft-boutros-mpls-tp-performance-01	Performance Monitoring of MPLS Transport Profile LSP	Ind.
draft-bryant-mpls-tp-ach-tlv-01	Definition of ACH TLV Structure	Ind.
draft-ceccarelli-mpls-tp-p2mp-ring-00	P2MP traffic protection in MPLS-TP ring topology	Ind.
draft-fhbs-mpls-tp-cv-proactive-00	MPLS-TP Proactive Continuity and Connectivity Verification	Ind.

RFC / Internet-Draft	Title	Status
draft-fulignoli-mpls-tp-ais-lock-tool-00	MPLS-TP OAM Alarm Suppression Tools	Ind.
draft-helvoort-mpls-tp-rosetta-stone-00	A Thesaurus for the Terminology used in Multiprotocol Label SwitchingTransport Profile (MPLS-TP) drafts/RFCs and ITU-T's Transport NetworkRecommendations.	Ind.
draft-liu-mpls-tp-bnm-00	Multiprotocol Label Switching Transport Profile Backward Notify MessagePacket	Ind.
draft-mansfield-mpls-tp-nm-framework-00	MPLS TP Network Management Framework	Ind.
draft-martinotti-mpls-tp-interworking-01	Interworking between MPLS-TP and IP/MPLS	Ind.
draft-sprecher-mpls-tp-survive-fwk-01	Multiprotocol Label Switching Transport Profile Survivability Framework	Ind.
draft-weingarten-mpls-tp-linear-protection-01	MPLS-TP Linear Protection	Ind.
draft-yang-mpls-tp-ring-protection-analysis-00	Multiprotocol Label Switching Transport Profile Ring Protection Analysis	Ind.

Table 1: Internet Drafts on MPLS-TP (March 30, 2009)

4 OAM Tools for MPLS-TP

The MPLS-TP OAM tool set is currently under definition at the IETF and comprises the OAM features listed in Figure 3. The detailed requirements for the various OAM functions can be found in the related Internet Drafts listed in Table 1. The fundamental idea is that dedicated OAM packets are interspersed into the associated user traffic flows. These OAM packets are created and processed by maintenance end point. Maintenance intermediate points can also process these OAM packets and may collect data or raise alarms. The tools can be categorized in proactive OAM functions that are running all the time and on-demand monitoring functions.

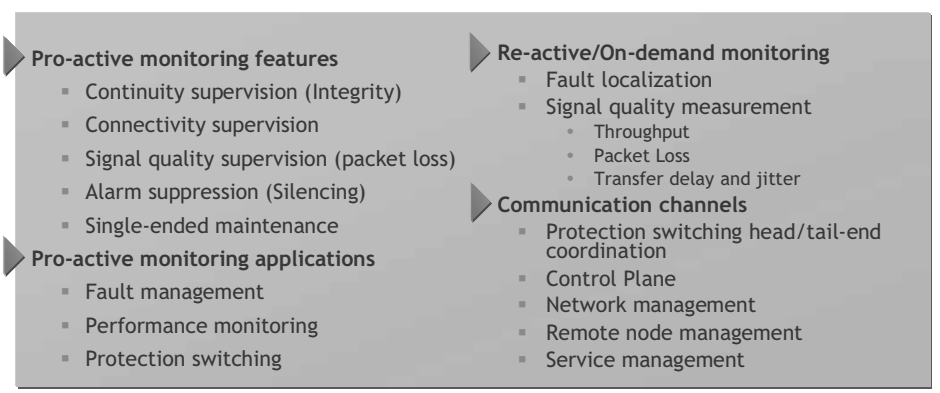


Figure 3: MPLS-TP OAM Tools

5 Control Plane for MPLS-TP

The IETF further defined Generalized MPLS (GMPLS) as a generalization of the MPLS control plane to develop a dynamic control plane that can be applied to packet switched and optical networks. The GMPLS architecture is described in [RFC3945]. The GMPLS control plane, or its ITU-T counterpart, Automatically Switched Optical Network (ASON) [G.8080], supports connection management functions as well as protection and restoration techniques and thus providing network survivability across networks comprising routers, MPLS-TP LSRs, optical ADMs, cross connects, and WDM devices.

MPLS-TP may utilize the distributed control plane to enable fast, dynamic and reliable service provisioning in multi-vendor and multi-domain environments using standardized protocols that ensure interoperability.

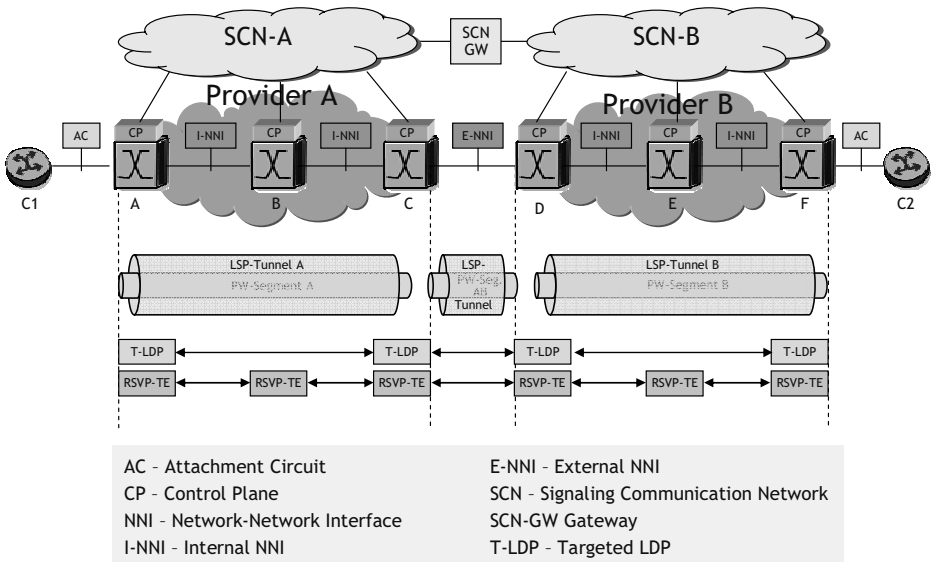


Figure 4: Control Plane View of a Multi-Segment Pseudowire

The MPLS-TP control plane is based on a combination of the MPLS control plane for pseudowires and the GMPLS control plane for MPLS-TP LSPs, respectively. This is illustrated in Figure 4. The distributed MPLS-TP control plane provides the following basic functions:

- Signaling
- Routing
- Traffic engineering and constraint-based path computation

Moreover, the MPLS-TP control plane is capable of performing fast restoration in the event of network failures.

The MPLS-TP control plane provides features to ensure its own survivability and to enable it to recover gracefully from failures and degradations. These include graceful restart and hot redundant configurations. The MPLS-TP control plane is as much as possible decoupled from the MPLS-TP data plane such that failures in the control plane do not impact the data plane and vice versa.

6 Synchronization in Packet Networks

SONET/SDH networks inherently provide synchronization whereas packet based network protocols like e.g. Ethernet are by nature asynchronous. To deploy an Ethernet based infrastructure for mobile backhauling, protocol extensions are required that provide these synchronization functions.

6.1 Clock Hierarchy

Starting at the Primary Reference Clock and ending at the clock in the node closest to the application we have a hierarchy of Master and Slave Clocks.

6.2 Synchronization Approaches

There are three different approaches to solve the synchronization issue:

1. An overlay synchronization network
2. A distributed reference clock solution
3. Forwarding of clock information across the packet domain

The overlay solution would require a synchronization network in parallel to the packet data network. In a distributed reference clock solution there is, at least at the edges of the packet network access to a primary reference clock, this could be provided by GPS. Forwarding clock information requires a synchronization protocol.

6.2.1 Packet Based Clock Recovery Solutions

There are two different clock recovery approaches:

1. Adaptive Timing
2. Differential Timing

6.2.1.1 Adaptive Clock Recovery (ACR)

In adaptive timing or adaptive clock recovery (ACR) the reference clock information is encapsulated and de-capsulated at the packet edge nodes that provide interworking function between TDM and packet domains:

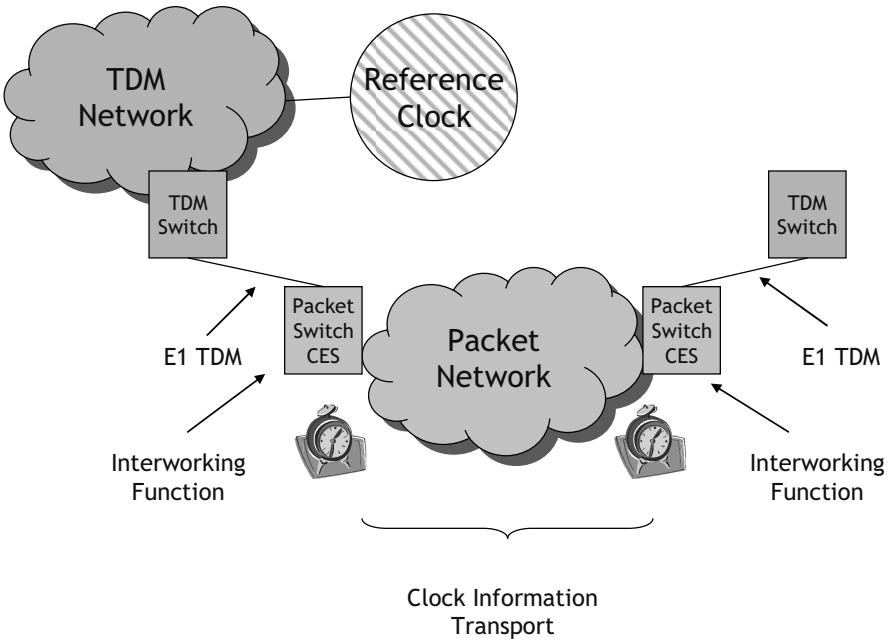


Figure 5: Clock Synchronization across a Packet Network

Here a protocol is required that regenerates at least frequency. This can be done by analyzing either inter packet arrival times or egress buffer fill grades. In general there are three different protocols in use today:

1. Network Time Protocol (NTP) according to RFC 1305; NTP is sufficient for OAM functions as it is precise within a range of hundreds of microseconds.
2. Proprietary implementations derived from NTP; these sometimes are sufficient for mobile backhauling. The number of nodes between master and slave is important here
3. Precision Time Protocol (PTP) according to IEEE 1588v2; this protocol recommends hardware generated timestamps. It is possible to transfer time of day and precise frequency. Intermediate nodes do not need to be compliant, though the number of non-compliant nodes has a drastic impact on the performance achieved by the protocol. Intermediate compliant nodes can either have a transparent clock regenerating clock from one slave port towards one master port or a boundary clock with multiple master ports.

6.2.1.2 Differential Clock Recovery

In differential clock recovery (DCR) both edge elements performing the interworking function have to have access to a common reference clock. Hence frequency is not calculated from the time interval between incoming packets. Still, time stamp based time of day delivery has to be taken into account.

6.2.2 Hardware based Frequency Distribution in Packet Networks

Like in SDH networks frequency can be distributed over a packet network independently of utilization. Under heavy network load packet based distribution of frequency will not always meet the stringent precision requirements of standards G.812 and G.813. In ITU standard G.8261 frequency transport on the physical layer of Ethernet is defined. The requirements for the node clocks are set in ITU standard G.8262. Since frequency distribution over Ethernet physical layer does not take into account time of day a combination with IEEE 1588v2 time stamping is the best way to implement Synchronous Ethernet.

Frequency distribution over Ethernet physical layer requires every node in the chain to adhere to ITU-T G.8262

6.3 Synchronization Status Messaging

To guarantee for SDH like redundancy in Synchronization distribution an additional protocol is required, that, in case of failing access to one PRC calculates the path to the secondary PRC. This SSM protocol does not have to have real time qualities since the equipment clocks can run independently from any PRC for a matter of days.

7 Physical Infrastructure Supporting MPLS-TP

It is mandatory for MPLS-TP that it can be carried over the existing and still evolving physical transport technologies such as SONET/SDH, OTN/WDM, and Gigabit Ethernet. The encapsulation techniques for these technologies are briefly described below.

7.1 MPLS-TP over SONET/SDH, PDH and OTN

ITU-T Recommendation G.7041 [G.7041] defines a generic framing procedure (GFP) to encapsulate variable length payload of various client signals for subsequent transport over SONET/SDH, PDH, and OTN networks. The GFP header contains a User Payload Identifier (UPI) field for which values are defined that indicate that the carried protocol data unit is an MPLS packet. MPLS-TP uses that same UPI code point as MPLS. The OTN [G.709] includes a WDM network layer for the transport of a variety of OTN client signals. In the SONET/SDH case, virtual concatenation can be applied to form transmission pipes with larger capacities (n x 150 Mbit/s).

6.2 MPLS-TP over Gigabit Ethernet

Similar to GFP, MPLS-TP can be carried across Ethernet links. A two-octet Ether Type field has been defined by the Ethernet II framing networking standard to indicate which protocol is encapsulated in the payload area of the frame.

7 Conclusions

MPLS-TP is intended to enable next-generation converged packet networks tying together service routing and transport platforms. Major advantages are consistent operations and OAM functions across the different network layers and the seamless interworking with IP/MPLS networks. MPLS-TP is highly scalable due to its multiplexing capability that can be used to create a network with multiple hierarchical layers. MPLS-TP supports a huge variety of services that are encapsulated into pseudowires and it can be carried over the existing and evolving transport network infrastructure.

References

[G.709] ITU-T Recommendation G.709: "Interfaces for the Optical Transport Network (OTN)", March 2003

[G.783] ITU-T Recommendation G.783: "Characteristics of synchronous digital hierarchy (SDH) equipment functional blocks", March 2003

[G.811] ITU-T Recommendation G.811: "Timing Characteristics of Primary Reference Clocks", September 1997

[G.812] ITU-T Recommendation G.812: "Timing requirements of slave clocks suitable for use as node clocks in synchronization networks", June 2004

[G.813] ITU-T Recommendation G.813: "Timing characteristics of SDH equipment slave clocks (SEC)", March 2003

[G.872] ITU-T Recommendation G.872: "Architecture of optical transport networks", November 2001

[G.7041] ITU-T Recommendation G.7041: "Generic framing procedure (GFP)", October 2008

[G.8080] ITU-T Recommendation G.8080 and Amendment 1: "Architecture for the automatically switched optical network (ASON)", June 2006, March 2008

[G.8261] ITU-T Recommendation G.8261: "Timing and synchronization aspects in packet networks", April 2008

[G.8262] ITU-T Recommendation G.8262: "Timing characteristics of synchronous ethernet equipment slave clock (EEC)", August 2007

[G.8264] ITU-T Recommendation G.8264: "Timing distribution through packet networks", October 2008

[M.3010] ITU-T Recommendation M.3010: "Principles for a telecommunications management network", February 2000

[M.3400] ITU-T Recommendation M.3400: "TMN management functions", February 2000

[RFC1305] Mills, L.: "Network Time Protocol (Version 3)", IETF RFC1305, March 1992

[RFC3916] Xiao, X., McPherson, D., and Pate, P.: "Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)", IETF RFC3916, September 2004

[RFC3945] E.Mannie: "Generalized Multi-Protocol Label Switching (GMPLS) Architecture", IETF RFC3945, October 2004

[IEEE 1588v2] "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems", IEEE 1588v2, March 2008

Network Access Control

Michael Epah

Praktische System Sicherheit
Fraunhofer-Institut Sichere Informations-Technologie
Rheinstrasse 75
64295 Darmstadt
michael.epah@sit.fraunhofer.de

Abstract: Die Bedrohung der Informationssicherheit durch vermeintlich vertrauenswürdige mobile Endgeräte, die unkontrolliert an das Netzwerk angeschlossen werden, ist nicht zu unterschätzen. Unzureichend administrierte mobile Endgeräte können Schadprogramme „einschleppen" und so die zentralen Schutzmaßnahmen aushebeln. Deshalb ist es notwendig, dass Endgeräte vor dem Zugang zum Netzwerk überprüft werden und korrumpierte Systeme „unter Quarantäne gestellt" werden. Die Technik, durch die sichergestellt wird, dass Endgeräte nicht unkontrolliert in das Netzwerk kommen, nennt man „Network Access Control" (NAC). Die aktuellen auf dem Markt befindlichen NAC Produkte verfolgen unterschiedliche Ansätze. Es gibt auch Bemühungen für eine Standardisierung. Dieser Beitrag stellt die unterschiedlichen Ansätze von NAC vor und gibt IT-Managern Hinweise für die Einführung von NAC. Abschließend wird der Versuch unternommen die Zukunftschancen der unterschiedlichen Ansätze zu bewerten. Die vorliegende Arbeit entstand aus einem Evaluationsprojekt zum Thema NAC im Jahr 2008. Dabei wurden NAC Produkte auf ihre Eignung für den Einsatz in der Fraunhofer-Gesellschaft untersucht.

1 Was ist „Network Access Control" (NAC)?

Wie der Name „Network Access Control" (NAC) zum Ausdruck bringt, geht es bei NAC in erster Linie um die Überprüfung von Benutzern und deren Endgeräte, bevor ihnen, entsprechend den Netzwerk-Sicherheitsrichtlinien des Unternehmens, Zugang ins Netzwerk gewährt, verweigert oder nur eingeschränkt erlaubt wird. Diese Richtlinien können z.B. besagen, dass alle Mitarbeiter sich gegenüber dem Windows Domain Controller anmelden müssen. Nach erfolgreicher Anmeldung wird das Endgerät des Mitarbeiters einem „Gesundheitscheck" unterzogen, um sicher zu stellen, dass die Schutzprogramme (Antivirus, Antispyware, Personal Firewall) auf dem aktuellen Stand sind und auch laufen. Darüberhinaus dürfen Windows 2000, XP und Vista nur mit den innerhalb des Unternehmens freigegebenen neuesten Patches benutzt werden. Benutzer ohne ein Windows Domain Account werden als Gäste betrachtet und müssen sich über ein Web-Captive Portal anmelden. Gäste bekommen Internet-Zugang mittels eines Gast-Accounts, aber keinen Zugang zum internen Netzwerk. Mitarbeiter-Endgeräte, die den „Gesundheitscheck" nicht bestehen oder sich verdächtig im Netzwerk verhalten kommen in einen separaten Netzwerk-Bereich, von dem aus nur bestimmte Server erreichbar sind.

1.2 Ablauf eines NAC-Prozess

Ein Benutzer schließt ein Endgerät, EP1, über LAN, WLAN oder VPN an das Netzwerk an. Als erstes geschieht „Endpoint Detection" – die NAC-Lösung bemerkt EP1. Danach folgt die Feststellung der Identität des Benutzers durch einen Login-Vorgang (Benutzer-Authentifizierung). Als nächstes wird der Zustand des Endgerätes überprüft („Endpoint Assessment"). Benutzer-Authentifizierung und Endpoint Assessment werden auch Pre-Connect bzw. Pre-Admission Assessment genannt. Was genau zu überprüfen ist, wird von den Netzwerk-Sicherheitsrichtlinien des Unternehmens bestimmt. Endgeräte, die die Überprüfung nicht bestehen, werden ganz oder teilweise „unter Quarantäne" gestellt. Muss ein Endgerät unter Quarantäne gestellt werden, wird der Benutzer informiert und in der Behebung („Remediation") der gefundenen Sicherheitsmängel unterstützt. War Pre-Connect Assessment erfolgreich, bekommt das Endgerät vollen Netzzugang und bleibt, je nach NAC-Produkt, unter Beobachtung - Post-Connect Assessment genannt.

1.3 Betriebsmodus einer NAC-Lösung

Wenn ein NAC-Server, wie in Abbildung 1, so platziert ist, dass sämtlicher Datenverkehr von und zu dem Endgerät immer durch den NAC-Server geht, spricht man von einer „**in-band-Lösung**". Vorteilhaft hierbei ist, dass Netzwerkomponenten (Switches, Hubs, Access Points, usw.) beliebiger Hersteller unterstützt werden können, weil die Durchsetzung der Netzwerk-Sicherheitsrichtlinien am NAC-Server geschieht und nicht an den Netzwerkkomponenten. In-Band-Lösungen können auch leichter Intrusion Detection Funktionalität integrieren, da der Datenverkehr auch nach dem Pre-Connect-Assessment durch den NAC-Server geleitet wird. Nachteilig ist jedoch, dass der NAC-Server einen „single point of failure" darstellt und die Netzwerkstruktur so verändert werden muss, dass der gesamte Datenverkehr über den NAC-Server geht. Außerdem bleiben compliant und non-compliant Endgeräte in demselben Netzbereich, so dass compliant-Clients von den non-compliant Clients angegriffen werden können.

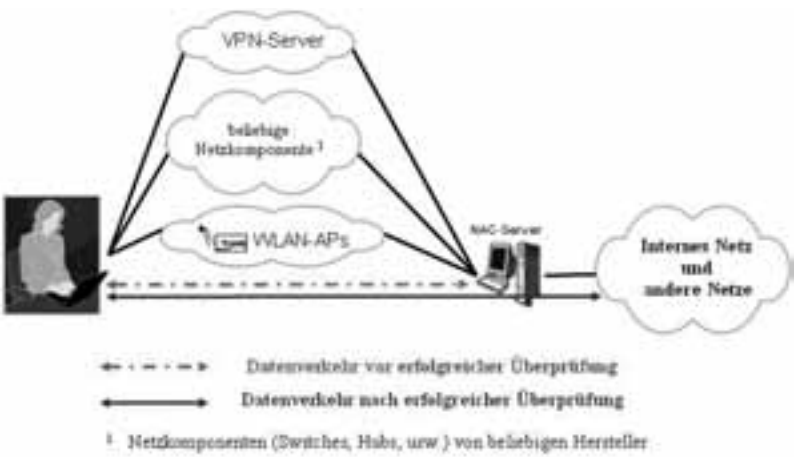


Abbildung 1: NAC-Server in **in-band** Modus

Die zweite Möglichkeit ist, den NAC-Server so zu platzieren, dass sämtlicher Datenverkehr von und zu dem Endgerät nur während der Überprüfung der Richtlinienkonformität durch den NAC-Server geht. Nach erfolgreicher Überprüfung geht der Datenverkehr von und zu dem Endgerät nicht mehr über den NAC-Server. In diesem Fall spricht man davon, dass der NAC-Server „**Out-of-band**“ ist. Dieses Szenario ist in Abbildung 2 dargestellt. Die Nachteile von „in-band“ Lösungen entfallen bei Out-of-Band-Lösungen. Ein großer Nachteil von Out-of-Band-Lösungen ist aber, dass nur bestimmte Produkte ausgewählter Hersteller unterstützt werden. Außerdem ist die Integration von Intrusion Detection Funktionalität schwieriger.

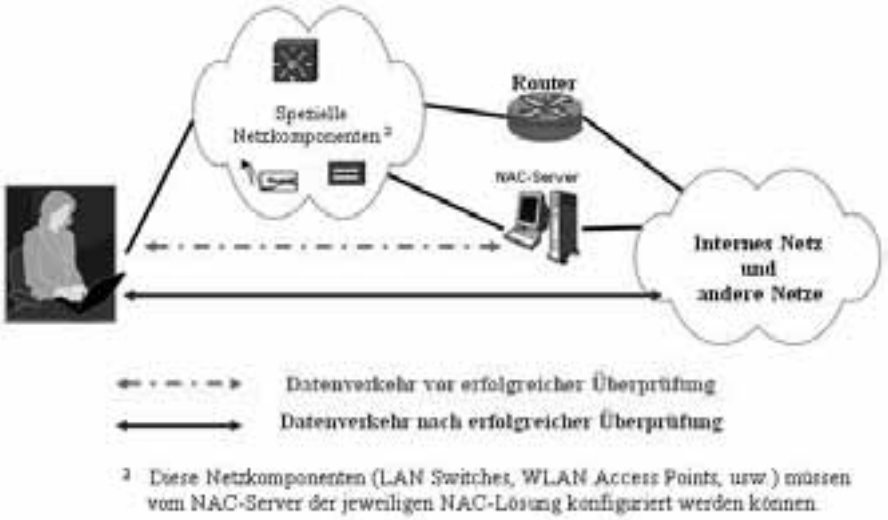


Abbildung 2: NAC-Server im **out-of-band** Modus

2 Implementierungswege von NAC-Grundfunktionen

2.1 Endpoint Detection

Bevor eine NAC-Lösung Endgeräte überprüfen kann, muss sie diese aufspüren, sobald sie im Netzwerk auftauchen - unabhängig davon über welchen Weg (LAN, WLAN, VPN) das Endgerät Netzwerkzugang bekommen hat. Das Aufspüren von Endgeräten kann durch verschiedene Wege erreicht werden. Je mehr von den folgenden Möglichkeiten ein NAC-Produkt unterstützt, um Endgeräte aufzuspüren, desto besser.

IEEE 802.1X: Bei dieser Methode werden die LAN Switches bzw. WLAN Access Points so konfiguriert, dass der Authentifizierungs-Verkehr über den NAC-Server geht.

DHCP Detection: Bei dieser Methode werden Endgeräte aufgespürt, wenn sie DHCP-Anfragen starten. Sie kann umgangen werden, wenn Endgeräte kein DHCP verwenden.

„Sniffen“ von Netzwerkverkehr: Bei dieser Methode werden durch Sniffen von Netzwerkverkehr neue Endgeräte im Netzwerk bemerkt. Falls nicht nur Broadcast sondern jeder Netzwerkverkehr von allen Endgeräten beobachtet wird, kann diese Methode nicht umgegangen werden. Dazu muss an einem geeigneten Punkt im Netzwerk der gesamte Verkehr zu einem Netzwerk-Anschluß gespiegelt werden.

Periodisches Abfragen (Polling) von Netzwerk-Komponenten: Netzwerk-Komponenten, wie Switches und Router pflegen Tabellen von Endgeräten, die über sie Zugang haben. Diese Tabelle werden periodisch, etwa alle 5 Minuten, abgefragt. Nachteilig ist die relativ lange Zeit, die vergeht, bevor ein Endgerät aufgespürt wird.

SNMP Traps: Einige Netzwerk-Komponenten können so konfiguriert werden, dass sie das Auftauchen neuer Endgeräte durch SNMP Traps an vorgegebene Server melden.

2.2 Benutzer- bzw. Endgeräte-Authentifizierung

In vielen Netzwerken werden Benutzer gegenüber Active Directory, LDAP oder RADIUS Server authentifiziert. Die Fähigkeit Single-Sign-On (SSO) durch „snooping“ bzw. „sniffing“ (Abhören) von Anmeldeverkehr zu ermöglichen, erhöht die Akzeptanz des Produkts bei Benutzern. Je mehr von den folgenden Benutzer-Authentifizierungs-Möglichkeiten eine NAC-Lösung anbietet, desto besser.

IEEE 802.1x mit EAP: Der Netzwerkzugangs-Switch bzw. -WLAN Access Point sorgt dafür, dass das Endgerät nur mit dem Authentifizierungsserver kommunizieren kann, bis der Benutzer sich gegenüber dem Authentifizierungsserver authentifiziert hat. Der Vorteil dieser Methode ist, dass sie standardbasiert ist.

Web Captive Portal: Der Vorteil dieser Methode ist die Unabhängigkeit vom Betriebssystem des Endgeräts. Allerdings muss der Benutzer einen Webbrowser starten.

„MAC-authentication-bypass“: Endgeräte ohne Benutzer oder 802.1X Supplicant können nur anhand ihrer MAC-Adresse authentifiziert werden. Diese sind fälschbar, was ergänzende Maßnahmen, wie den Einsatz von Access Control Lists, erfordert.

Herstellerspezifische Methoden: Eine auf dem Endgerät installierte Software übernimmt die Logon-Daten und ermöglicht somit u.a. auch Single-Sign-On (SSO). Manche NAC-Lösung beobachten („sniffen“) die Login-Session zwischen Endgerät und Authentifizierungsserver, um SSO zu ermöglichen. Schlägt der Login fehl oder findet er überhaupt nicht statt, muss der Benutzer sich über ein Webportal anmelden.

2.3 Überprüfung des Endgeräts (Endpoint Assessment)

Bei Endpoint Assessment geht es darum zu überprüfen, ob alle vorgeschriebenen Schutzprogramme wie Antivirus, Antispyware, Personal Firewall usw. sowie das Betriebssystem des Endpoints uptodate und aktiviert sind.

Darüber hinaus soll dadurch sichergestellt werden, dass auf dem Endgerät nur die erlaubten Services bzw. Programme aktiv sind. Es gibt zwei grundsätzliche Varianten für Endpoint-Assessment. Bei **„Agent based"-Assessment** läuft ein eigenständiges Programm auf dem Endgerät. Das kann ein „persistent Agent", eine fest installierte Software auf dem Endgerät, oder ein „dissolvable Agent", welcher immer wieder herunter auf das System geladen werden muss, sein. „Agent based"-Lösungen unterstützen oft nur bestimmte Betriebssysteme und können Gäste-Endgeräte nicht überprüfen. **„Agentless"-Assessment** kann auf einem Java Applet, das im Browser auf dem Endgerät ausgeführt wird, basieren. Das Scannen eines Endgeräts von „außen" mittels Netzwerkscanner, wie Nmap, zählt auch zu **„Agentless"-Assessment.** Einige Hersteller setzen aber auch Scanner ein, die per Net-logon sich mit dem Endgerät verbinden, um lokale Tests durchzuführen. Ein entsprechender Account wird dafür benötigt. Die dritte **„Agentless"-Assessment** Variante beobachtet und analysiert den Verkehr des Endgerätes wie ein Intrusion Detection System (IDS), um festzustellen, ob verbotene oder infizierte Software auf dem Endgerät aktiv ist

2.4 Durchsetzung der Netzwerk-Sicherheitsrichtlinien (Enforcement)

Bei diesem Teilaspekt von NAC wird danach gefragt, wie das NAC-Produkt die nicht richtlinienkonformen Endgeräte unter Quarantäne stellt und wo im Netzwerk das Durchsetzen (Enforcement) der Richtlinien erfolgt. Enforcement geschieht generell entweder auf dem Endgerät selbst, oder auf Komponenten (Switches, Access Points, VPN-Server, NAC Appliances) im Netzwerk. Nachteilig bei End-Point-Enforcement ist, dass diese Methode nur auf Endgeräten möglich ist, die unter der Kontrolle des Administrators sind. Enforcement direkt am Netzwerkzugangspunkt, wie an Switches, ist am effektivsten. Gebräuchliche Methoden und Orte der Durchsetzung sind:

„VLANs": Das VLAN des Endgeräts wird mittels IEEE 802.1X oder Command Line Interface (CLI) Befehle verändert. Nachteilig hierbei ist, dass die Endgeräte zwischen IP-Subnetzen hin und her verschoben werden. Außerdem können Endgeräte im Quarantäne-VLAN sich gegenseitig angreifen oder infizieren.

„Access Control Lists (ACLs)": Bei L2-ACLs werden Zugriffsrechte eines Endgeräts direkt an dem Netzwerkzugangspunkt kontrolliert. Nachteilig ist, dass ACLs entsprechende Netzwerkkomponenten erfordern. Vorteilhaft ist aber, dass Per-User-ACLs ein gegenseitiges Infizieren im Qurantäne-Netz verhindern können.

Firewalls. Filtering des Verkehrs durch eine Personal-Firewall auf dem Endgerät („Endpoint based NAC") oder durch eine Inband- bzw. „Virtual"-Firewall im Netzwerk.

MAC-Address-Filtering: Der Administrator trägt die zugelassenen MAC-Adressen ein.

ARP-Einträge des Endpoints: Durch Veränderung der ARP-Tabelle des Endpoints und der anderen Komponenten geht der Verkehr des Endgeräts über den NAC-Server.

„End Point Routing": Die Routing-Tabelle des Endpoints wird so verändert, dass nur bestimmte Ziele erreicht werden können.

3 NAC-Architectures

3.1 Trusted Computing Group‘s Trusted Network Connect (TCG-TNC) [1]

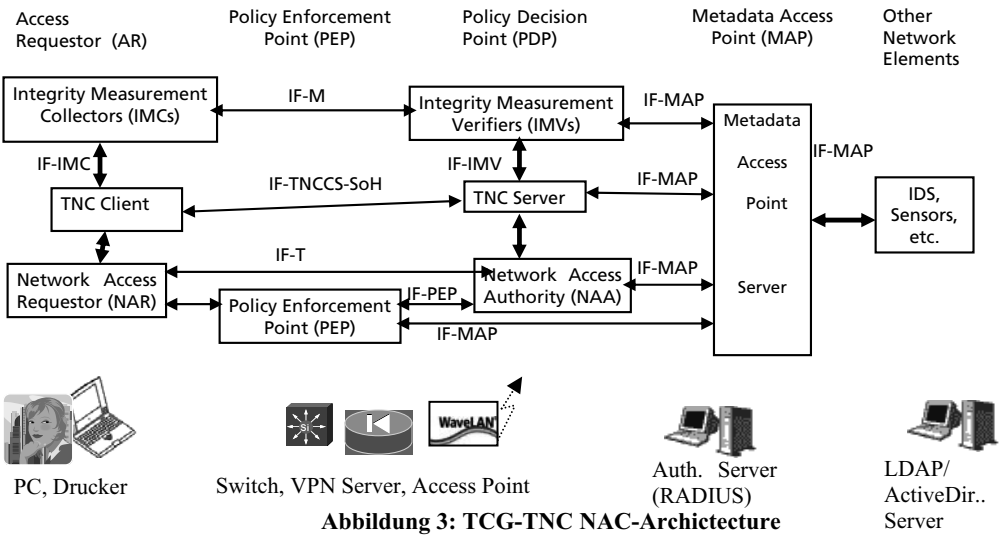


Abbildung 3: TCG-TNC NAC-Archictecture

Die erste Säule der TCG-TNC Architecture ist der „Access Requestor“, das Endgerät, das Netzzugang sucht. Netzwerkkomponenten, über die das Endgerät Netzwerkzugang bekommen soll und die in der Lage sind, so gesteuert zu werden, dass das Endgerät ganz, teilweise oder nur eingeschränkt Zugang bekommt, bilden die Policy Enforcement Point (PEP) Säule. Das können LAN Switches, WLAN Access Points oder VPN Server sein. Die dritte Säule des TCG-TNC Models ist der Policy Decision Point (PDP). Die vierte Säule im TCG-TNC Model macht dieses Model zu dem umfassendsten Model zurzeit. Der „Metadata Access Point“ (MAP) ermöglicht das Einfließen von Ergebnisse anderer Netzwerküberwachungstools wie IDS-Systemen und anderen Sensoren in die Entscheidung darüber, ob ein Endgerät vollen, begrenzten oder gar keinen Zugang bekommen bzw. weiterhin bekommen darf. Das erleichtert bzw. ermöglicht in manchen Fällen erst das Post-Connect-Assessment - auch Continuous Assessment genannt.

Stand von Implementierungen
Die Kommunikation zwischen einem Access Requestor (AR) und PEP (siehe Abbildung 3) läuft über vorhandene Protokolle, wie IEEE 802.1X oder IPSec IKEv2. Das IF-PEP Protokoll verwendet das RADIUS Protokoll und sieht drei Möglichkeiten vor: vollen, keinen oder durch VLANs bzw. Access Control Lists eingeschränkten Netzwerkzugang. Das IF-T Protokoll verwendet das Tunneled EAP Protokoll. Das IF-TNCCS-SoH Protokoll ist kompatibel mit der NAC-Lösung von Microsoft. Bei den Protokollen IF-M und IF-MAP verwenden Produkte zurzeit proprietäre Methoden.

3.2 Microsoft Network Access Protection (MS-NAP) [2]

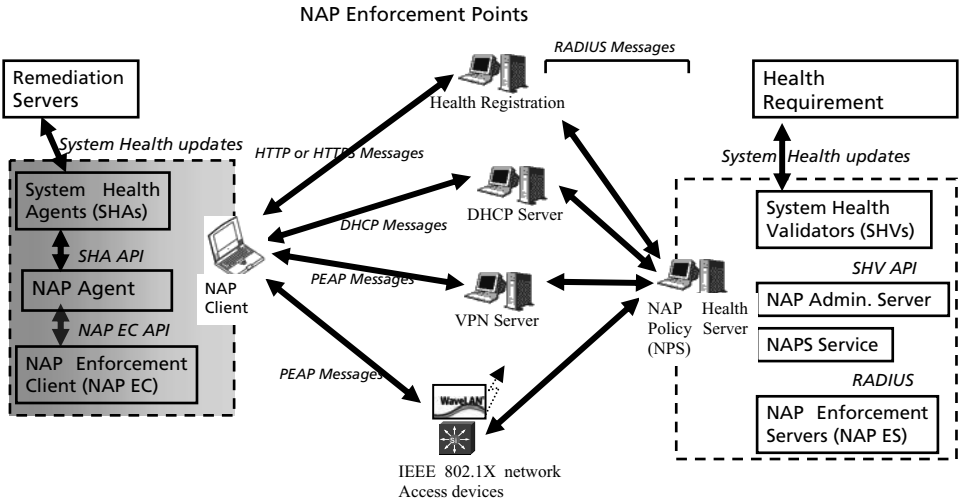


Abbildung 4: Microsoft NAP-Archictecture

Die Microsoft-NAP Architektur hat drei Säulen: der NAP Client, entsprechende Enforcement Points und den NAP Health Policy Server (NPS). Drittanbieter können Komponenten, wie SHA mit dem entsprechenden NAP EC, NAP ES und SHV liefern. Zurzeit liefert Microsoft NAP Enforcement Clients für IPSec, DHCP, VPN, und IEEE 802.1X Enforcement. Durch die offenen APIs gibt es bereits Produkte von Drittanbietern für die Integration der Linux / Unix Welt. Ein großer Schwachpunkt von MS-NAP ist Client Detection in einem Netzwerk ohne IEEE 802.1X Unterstützung. Nachteilig ist auch dass PDAs, Drucker, usw. nicht unterstützt werden.

3.3 NAC-Architectures - IETF Network Endpoint Assessment (IETF NEA) [3]

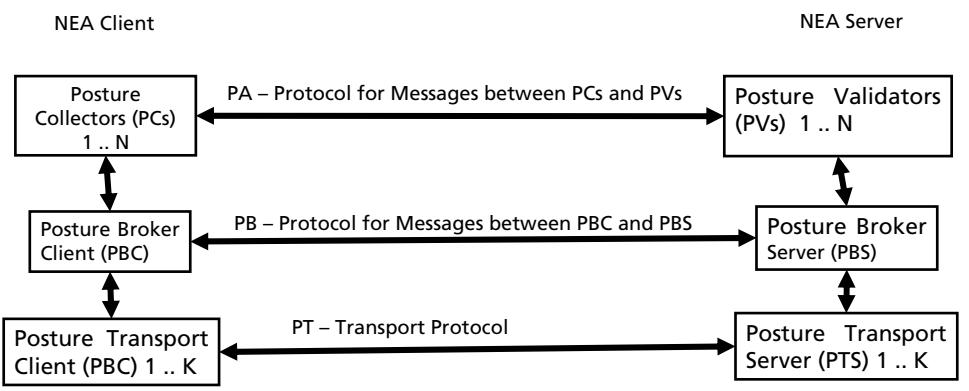


Abbildung 5: IETF NEA Archictecture

Die IETF-NEA Gruppe erfüllt zurzeit zwei wichtige Aufgaben: Erstens die Normierung der Begriffe im NAC Umfeld (Tabelle 1) sowie Bewertung guter Industrie-Standards, insbesondere von TCG, um diese eventuell als IETF Standards zu übernehmen.

IETF-NEA	**TCG-TNC**	**MS-NAP**	**Kommentar**
NEA Client (Software auf einem) Endpoint	Access Requestor	NAP Client	Ein Bündel von Software, die es einem Endgerät (PC, usw.) ermöglicht mit einer NAC-Lösung zusammen zu arbeiten. Der Begriff bezeichnet in der Praxis auch den Endpoint selbst.
Posture Collectors	Integrity Measurement Collector	System Health Agent	Eine oder mehrere Komponenten eines NEA Clients, welche den Ist-Zustand eines oder mehrerer Aspekte (z.B. Antivirus, Host IPS, usw.) eines Endpoints ermitteln und weiterleiten.
Posture Broker Client	TNC Client	NAP Agent	Eine Komponente eines NEA Clients, welche die Ist-Zustände einiger Aspekte des Endpoints von den entsprechenden Posture Collectors sammelt, weiterleitet, und eventuelle „Reparatur-Massnahmen“ an Posture Collectors zurück gibt.
Posture Transport Client	Network Access Requestor	NAP En-forcement Client	Eine Software-Komponente auf dem Endpoint, mit dem der Netzzugang realisiert wird. Beispiele sind IEEE 802.1X Supplicant, VPN Client, usw.
NEA Server	Policy Decision Point	NAP Health Policy Server	Ein Bündel von Software, die es einem Gerät(„NAC-Server“) im Netzwerk ermöglicht, die Ist-Zustände von Endpoints auf ihre Konformität zu den vorgegebenen Netzwerk-Sicherheitsrichtlinien hin zu überprüfen und entsprechende Maßnahmen zu veranlassen.
Posture Validator	Integrity Measurement Verifier	System Health Validator	Software auf einem NEA Server, die den übermittelten Ist-Zustand mit dem Soll-Zustand vergleicht und ein entsprechendes Ergebnis liefert.
Posture Transport Server	Network Access Authority	NAP En-forcement Server	Software Komponente des NEA Servers passend zu entsprechenden Komponenten des NEA Clients.
Inter-mediary devices	Policy Enforcement Point	NAP En-forcement Points	Netzknoten mit der Fähigkeit Maßnahmen des NEA Servers durchzusetzen. Die Funktionalität dieser Geräte ist nicht Bestandteil von IETF NEA.

Tabelle 1: Übersicht über einige NAC-Begriffe

4 Zukunfts-Chancen verschiedener NAC-Ansätze

Es gibt, wie im Kapital 2 beschrieben, verschiedene Wege der Implementierung von NAC-Funktionen. Zunächst wird eine NAC-Lösung, um am Markt zukünftig bestehen zu können, TCG-TNC konform sein müssen, da selbst die IEFT-NEA Gruppe Teile davon übernimmt [4] [5]. Darüber hinaus werden zwei Problemfelder, umfassende Endpoint Detection und „lying Endpoints“, letztlich die „Überlebensfähigkeit“ von NAC-Lösungen bestimmen.

4.1 Endpoint Detection als mögliches KO-Kriterium

Bei Endpoint Detection erweist sich IEEE 802.1X als die einzige Methode, die sowohl herstellerneutral als auch absolut zuverlässig ist. In Zukunft werden Endgeräte wie Drucker, Kameras, usw. IEEE 802.1x Supplicants haben müssen, wie das gerade bei VoIP Telefon geschieht. Solange es aber Endgeräte gibt, die IEEE 802.1X nicht unterstützen, wird Endpoint Detection mittels IDS/IPS Systemen die Methode der Wahl sein. Sie ist herstellerneutral und zuverlässig, obwohl nicht so zeitnah wie 802.1X. Endpoint Detection über DHCP-Server kann umgangen werden und Methoden, die einen Software-Agent auf den Endpoints voraussetzen, können Endpoints ohne solche Agenten nicht entdecken. Ein ungelöstes Problem bei Endpoint Detection sind Virtual-Machines die durch PAT (Port Address Translation) sowohl die IP-Adresse als auch die Mac-Adresse des Host-Systems nutzen. Solche Virtual-Machines werden zurzeit nicht unterschieden von ihrem Host-System. IDS/IPS basierte Lösungen sind vom Ansatz her in der Lage erkennen zu können, ob mehrere Maschinen hinter einem IP-Mac-Adresse-Paar stecken. Eine NAC-Lösung muss angesichts der wachsenden Beliebtheit von Virtualisierung dieses Problem beherrschen, um zukünftig bestehen zu können.

4.1 Entlarvung von „lying“ Endpoints als mögliches KO-Kriterium

Zurzeit sind alle NAC-Lösungen konfrontiert mit dem Problem von Endgeräten, die durch manipulierte Software falsche Angaben über sich selbst machen („lying Endpoints“) [6]. TCG-TNC sieht vor mittels TPM (Trusted Platform Modul) solche Software-Manipulationen zu erkennen [7]. Das wird bei Pre-Connect Assessment wahrscheinlich in sehr streng kontrollierten Umgebungen funktionieren. Für die meisten, wenn nicht sogar alle Umgebungen wird das Problem des „lying Endpoints“ nur durch IDS/IPS basierte NAC-Lösungen in den Griff zu bekommen sein. Auf jeden Fall werden auf lange Sicht nur solche NAC-Lösungen „überleben“, die überprüfen können, ob das Verhalten des Endpoints im Netzwerk zu den Angaben des Endpoints über sich selbst passt. Lösungen, bei denen Enforcement durch das Endgerät geschieht, haben aus diesem Grund schlechte Zukunftschancen. Die Anbieter von NAC-Lösungen scheinen die Wichtigkeit von IDS/IPS erkannt zu haben. Anbieter, die nicht aus der IDS/IPS Umgebung kommen, versuchen durch Kooperationen mit IDS/IPS Lösungen ihre Lösungen zu ergänzen. Es bleibt jedoch die Frage, ob der Kunde bereit sein wird zwei Produkte zu kaufen, die er selbst integrieren muss oder nicht lieber gleich ein Produkt kaufen wird, das bereits integriert ist.

5 Getting started

Ein Unternehmen sollte bei der Einführung von NAC darauf achten, dass das gewählte Produkt folgende Kriterien erfüllt

Sehr zu empfehlen sind Produkte bei denen eine einzige Appliance ausreicht, um mit dem Einsatz von NAC zu beginnen. Das Produkt sollte die vorhandenen Netzwerkkomponenten (Switches, Router, usw.) sowie die vorhandenen Authentifizierungsmethoden unterstützen. Ideal ist, wenn sich die Benutzer nicht anders authentifizieren müssen als vor der Einführung der NAC-Lösung. Das Produkt sollte eingesetzt werden können, ohne die Netzwerkstruktur zu verändern. An diesem Kriterium scheitern die meisten In-Band-Lösungen. Die Unterstützung von einem „monitor-only"-Mode ermöglicht es vorher zu sehen, wieviele und welche Geräte und Benutzer vorher angepasst werden müssen. Und schließlich sollte das Produkt eine schrittweise Einführung ermöglichen. Der Netzwerkadministrator hat dadurch die Möglichkeit die Erzwingung der Sicherheitsrichtlinien erst bei einigen ausgewählten Benutzern und Endgeräten zu aktivieren. Missglückt etwas dabei, wird es nur eine sehr kleine und überschaubare Gruppe betreffen. Diese Gruppe sollte so gewählt werden, dass sie für ihre Arbeit nicht auf eine dauernde Verfügbarkeit des Netzwerks angewiesen ist.

Literaturverzeichnis

[1] Trusted Computing Group, TNC Architecture for Interoperability Specification Version 1.3, Revision 6, 28 April 2008.
https://www.trustedcomputinggroup.org/specs/TNC/TNC_Architecture_v1_3_r6.pdf

[2] Microsoft Corporation, Network Access Protection Platform Architecture, June 2004, updated February 2008.
http://www.microsoft.com/technet/network/nap/naparch.mspx

[3] P. Sangster, H. Khosravi, M. Mani, K. Narayan, J. Tardo, Network Endpoint Assessment (NEA): Overview and Requirements, RFC 5209, Informational, June 2008. IETF. http://www.ietf.org/rfc/rfc5209.txt

[4] P. Sangster, K. Narayan, PA-TNC: A Posture Attribute Protocol (PA) Compatible with TNC, draft-ietf-nea-pa-tnc-03.txt, proposed standard, March 2009. IETF. http://www.ietf.org/internet-drafts/draft-ietf-nea-pa-tnc-03.txt

[5] R. Sahita, S. Hanna, R. Hurst, K. Narayan, PB-TNC: A Posture Broker Protocol (PB) Compatible with TNC, draft-ietf-nea-pb-tnc-03.txt, proposed standard, March 2009. IETF. http://www.ietf.org/internet-drafts/draft-ietf-nea-pb-tnc-03.txt

[6] Michael Thumann, Dror-John Röcher, NAC@ACK: Hacking the Cisco NAC Framework, March 2007.
http://www.ernw.de/content/e7/e181/e566/download568/ERNW_nacattack_10_dr_20070307_ger.pdf

[7] TCG Specification Architecture Overview, Specification Revision 1.4, August 2007.
https://www.trustedcomputinggroup.org/groups/TCG_1_4_Architecture_Overview.pdf

Messen, Analysieren und Überwachen im Rechenzentrum

Statistische Analyse von Delay-Messungen zur Performance-Evaluation in Netzwerken

Thomas Holleczek

holleczek@ife.ee.ethz.ch

Abstract: Die Messung von Paketlaufzeit und Paketverlust durch dedizierte Testpakete in Computernetzwerken ermöglicht die Beurteilung der Dienstgüte auf dem Pfad, den diese durchlaufen. Aus diesem Grund werden im X-WiN, dem Deutschen Forschungsnetz, und GÉANT2, seinem europäischen Ebenbild, seit vielen Jahren vom WiN-Labor der Universität Erlangen-Nürnberg auf sämtlichen Strecken IP Performance-Messungen durchgeführt. Dieses Paper gibt einen Überblick darüber, wie die Ergebnisse dieser Messungen durch statistische Methoden analysiert werden können, um mehr über den Zustand und die Performance der überwachten Netzwerk-Strecken zu erfahren.

1 Einleitung

Für viele Multimediaanwendungen wie VoIP oder Videokonferenzen ist ein hoher Dienstgüte-Standard unabdingbar. Deshalb entwickelt das WiN-Labor an der Universität Erlangen-Nürnberg seit 1997 das aktive Messsystem Hades, mit Hilfe dessen sich IP Performance Metrics (IPPM) wie One-Way Delay (OWD), IP Delay Variation (IPDV) und One-Way Packet Loss (OWPL) in Computernetzwerken bestimmen lassen.

Statistische Analysen der durch Hades erzeugten Messwerte bilden nun die Grundlage eines in der Entwicklung stehenden Alarmsystems, das in der Lage ist, Verschlechterungen der Netzwerk-Performance automatisch zu erkennen und entsprechende Warnungen zu generieren.

2 Hades-Messsystem

2.1 Hardware

Das Messsystem besteht aus Intel® Pentium® 4 PCs mit Fedora Core als Betriebssystem, die an alle wesentlichen Router im X-WiN und GÉANT2 angeschlossen sind. Jede dieser Messboxen ist mit einer GPS-Karte ausgestattet, um für eine Messgenauigkeit von mindestens 7 Mikrosekunden zu sorgen.

2.2 Software

Die Hades-Software besteht aus einem Sende- und einem Empfangsprozess, von denen jeweils eine Instanz auf jeder Messbox läuft.

Der Sendeprozess auf dem Quellrechner einer Messung generiert UDP-Messpakete und versieht diese mit einem Zeitstempel. Zusätzlich beinhaltet der Paket-Header eine eindeutige Sequenznummer, um den Verlust oder Vertauschung von Testpaketen erkennbar zu machen.

Auf dem Empfangsrechner einer Messung nimmt der entsprechende Empfangsprozess eintreffende Pakete entgegen, versieht sie ebenfalls mit einem Zeitstempel und speichert diesen mit der Sequenznummer und dem Startzeitstempel in einer lokalen Datei. Danach stehen die Messdaten zur Auswertung bereit.

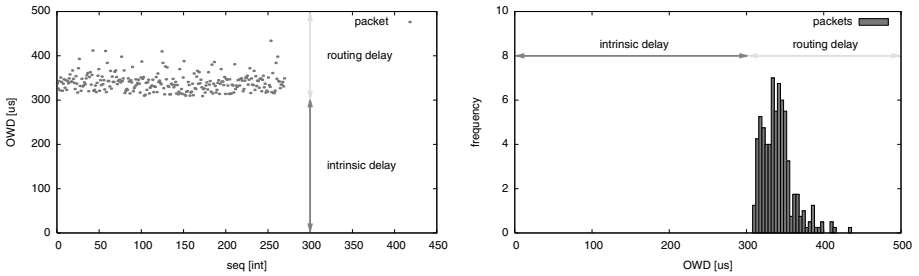
Die Messungen finden vollvermascht statt, so dass Statusinformationen für jede beliebige Strecke vorhanden sind. Derzeit generiert ein Sendeprozess für jeden bekannten Empfangsrechner alle 60 Sekunden einen Burst von 9 Testpaketen der Größe 41 Bytes. Der Abstand zwischen zwei aufeinander folgenden Paketen eines Bursts beträgt 30 Millisekunden, um Kollisionen an der Netzwerkschicht zu vermeiden.

3 Definitionen

Jede Strecke, d.h. jedes Paar von Quell- und Zielmessbox, zeigt im Bezug auf den gemessenen OWD ihr eigenes Verhalten. Die statistische Analyse betrachtet die in einem Zeitintervall zwischen zwei Messboxen gemessenen OWD-Werte $\mathbf{Y} = (y_1, \ldots, y_n)$. Wie in Abbildung 1 illustriert, wird der OWD offensichtlich durch zwei Komponenten beeinflusst:

1. **Intrinsic Delay.** Der *intrinsic delay* c zwischen zwei Messboxen wird dominiert durch die Ausbreitungsverzögerung, d.h. diejenige Zeit, die das elektrische bzw. optische Signal benötigt, um die Links des IP-Pfades zu durchlaufen, sowie die Übertragungszeit des Testpakets. Normalerweise ist c zwischen zwei Hosts konstant, lediglich im Falle einer Veränderung des durchlaufenen Pfades kommt es zu Veränderungen. Der *intrinsic delay* darf also als konstante untere Grenze betrachtet werden, die kein OWD-Wert unterschreiten kann.

2. **Routing Delay.** Zusätzlich besteht der OWD aus einem variablen Anteil, der auf das Verhalten der Router zurückzuführen ist, die die beiden Messboxen verbinden. Er wird als *routing delay* bezeichnet, da es sich bei diesem um diejenige Zeitspanne handelt, die alle involvierten Router benötigen, um das Testpaket zu verarbeiten. Der *routing delay* stellt also den interessanten Bestandteil des OWD dar, da dessen Analyse Rückschlüsse auf den dynamischen Zustand der Router erlaubt. Die entsprechende Folge von Werten des *routing delay* wird mit $\mathbf{X}$ gekennzeichnet und



(a) Sequentiell aufgezeichneter OWD (b) Entsprechendes Histogramm des OWD

Abbildung 1: Innerhalb einer halben Stunde zwischen zwei beliebigen Messboxen im X-WiN aufgezeichneter OWD

lässt sich in Abhängigkeit des OWD und des *intrinsic delay* angeben:

$$\mathbf{X} = (x_1, \ldots, x_n) = (y_1 - c, \ldots, y_n - c) \tag{1}$$

Der *routing delay* $\mathbf{X}$ kann also durch die Subtraktion des *intrinsic delay* c vom gemessenen OWD berechnet werden. Da c prinzipiell unbekannt ist, richtet die Vorverarbeitung der Messwerte das Hauptaugenmerk auf dessen Bestimmung.

4 Vorverarbeitung

Bevor der in einem beliebigen Intervall gemessene OWD $\mathbf{Y}$ statistisch analysiert werden kann, muss dieser vorverarbeitet werden. Dies umfasst die Säuberung der Messwerte: aufgrund von Uhrenfehlern ungültige Werte müssen verworfen werden. Zusätzlich werden alle aufgetretenen Pfadveränderungen—die immer dann auftreten, wenn Router oder Links auf dem verbindenden IP-Pfad ausfallen—durch die Anwendung eines Clustering-Algorithmus erkannt. Das Wissen um ein solches aufgetretenes Umrouting erlaubt es schließlich Netzwerkadministratoren, entsprechende Gegenmaßnahmen einzuleiten.

Vorausgesetzt, dass keine Sprungstelle aufgetreten ist, wird im Anschluss der *intrinsic delay* als die untere Schranke des OWD bestimmt. Da die spätere statistische Analyse den *routing delay* modelliert, kann dieser nun gemäß Gleichung (1) berechnet werden.

5 Datenanalyse

Nach der Vorverarbeitung der Messwerte ist es die Aufgabe der Datenanalyse, diese durch ein mathematisches Modell zu beschreiben, aus welchem sich Rückschlüsse auf den Zustand der entsprechenden Strecke ziehen lassen. Hier geht der Analyseprozess nun davon aus, dass der *routing delay* innerhalb des zu analysierenden Zeitfensters von einer Wahrscheinlichkeitsdichte $f(x|\theta)$ mit dem Parametervektor θ erzeugt wurde. Aufbauend auf

dieser Hypothese wird eine Parameterschätzung durchgeführt, z.B. gemäß der *maximum-likelihood*-Methode oder dem Prinzip *expectation-maximization*. Im Zuge dieser wird auf der Basis der Werte $\mathbf{X}$ eine Schätzung $\hat{\theta}$ für θ berechnet, so dass $f(x|\hat{\theta})$ mit großer Wahrscheinlichkeit für die Erzeugung von $\mathbf{X}$ verantwortlich war. Durch eine Interpretation der Schätzung $\hat{\theta}$ kann schließlich mehr über den Status der entsprechenden Strecke erfahren werden.

5.1 Einfache Wahrscheinlichkeitsverteilungen

Gut geeignet zur Modellierung des *routing delay* ist die Gamma-Verteilung $\mathcal{G}$, charakterisiert durch die zwei Parameter α und β und die folgende Dichtefunktion:

$$\mathcal{G}(x|\alpha,\beta) = x^{\alpha-1} \cdot \frac{\beta^{\alpha} \cdot e^{-\beta x}}{\Gamma(\alpha)} \tag{2}$$

Eine einfachere Alternative mit nur einem Parameter σ^2 und ähnlicher Form ist die Rayleigh-Verteilung $\mathcal{R}$ mit der Wahrscheinlichkeitsdichte:

$$\mathcal{R}(x|\sigma^2) = \frac{x \cdot \exp\left\{-\frac{x^2}{2\sigma^2}\right\}}{\sigma^2} \tag{3}$$

5.2 Mischverteilungen

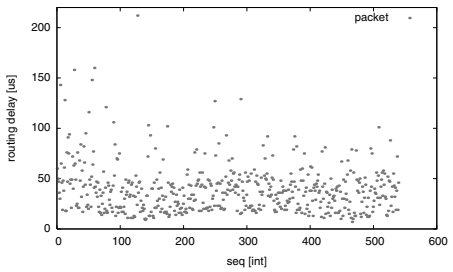
In einigen Situationen sind einfache Verteilungen zur Modellierung des *routing delay* jedoch nur bedingt geeignet—z.B. wenn das zu approximierende Histogramm wie in Abbildung 2 mehrere Peaks enthält. Hier ist es empfehlenswert, die Annahme zu machen, dass $\mathbf{X}$ von einer linearen Überlagerung von K gewichteten Gamma-Verteilungen, einer sog. Gamma-Mischverteilung (engl. *Gamma mixture model*), erzeugt wurde. Dies führt zu folgender Dichtefunktion:

$$f(x|\theta) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{G}(x|\alpha_k,\beta_k) \tag{4}$$

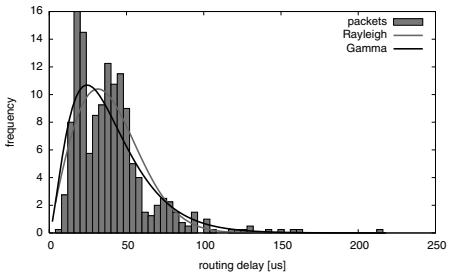
wobei π_k das Gewicht der k-ten Komponente ist.

5.3 Ergebnisse

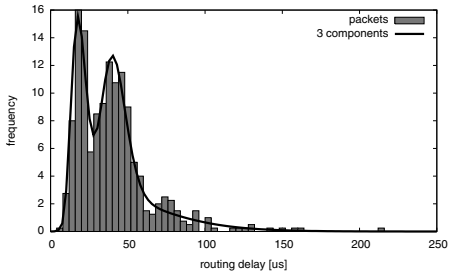
Die Adäquatheit der vorgestellten Modelle wurde im X-WiN für Zeitfenster der Breite 30 Minuten auf zahlreichen Strecken überprüft. Dabei konnten folgende Beobachtungen gemacht werden:



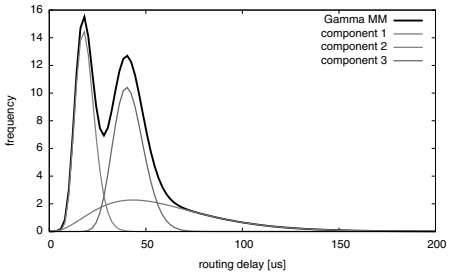
(a) *Routing delay*, der am 21. Oktober 2007 von 09:00 bis 10:00 Uhr zwischen GÉANT2-Messboxen in Thessaloniki und Wien aufgezeichnet wurde



(b) Aus dem Histogramm bestimmte einfache-Gamma und Rayleigh-Verteilung



(c) Gamma-Mischverteilung mit drei Komponenten, die aus dem Histogramm geschätzt wurde. Offensichtlich approximiert diese das Verhalten besser, da die Peaks des Histogramms klar erkannt werden.



(d) Verteilungen der drei Komponenten der Gamma-Mischverteilung. Die dritte Komponente (rot) modelliert den langsamen Abstieg des Histogramms nach rechts, was als Indikation für langsam entstehende Überlast interpretiert werden kann.

Abbildung 2: Verschiedene geschätzte Wahrscheinlichkeitsverteilungen

1. Die Gamma-Verteilung schneidet bei einer Evaluation aufgrund ihrer höheren Flexibilität leicht besser ab als die Rayleigh-Verteilung. Allerdings konnte nur in maximal der Hälfte aller untersuchten Fälle davon ausgegangen werden, dass diese einfachen Verteilungen als statistischer Generator des *routing delay* fungierten.

2. Bedingt durch die noch höhere Flexibilität können Gamma-Mischverteilungen die Form der beobachteten Histogramme wesentlich besser approximieren. Dies spiegelt sich in den Erfolgsquoten der Parameterschätzungen wieder: schon bei der Verwendung von drei Komponenten konnten in mehr als 80 Prozent aller Fälle die wiedergewonnen Gamma-Mischverteilungen als statistischer Generator des beobachteten *routing delay* angenommen werden. Darüber hinaus eignet sich dieses Modell auch zur frühzeitigen Erkennung entstehender Überlast einer Strecke: Diese äußert sich durch eine Komponente, die extrem langsam abfällt, wie in [Abb. 2d] illustriert.

Ein Beispieldiagramm mit dem *routing delay* einer Stunde sowie die aus diesem bestimmten Wahrscheinlichkeitsverteilungen findet sich in Abbildung 2.

6 Performance-Klassifizierung

In ausführlichen Analysen wurde das Verhalten sämtlicher Strecken im X-WiN untersucht. Dabei stellte sich heraus, dass sich die Performance einer Strecke im Hinblick auf den *routing delay* o.B.d.A. grob in vier Kategorien, die sog. Performance-Klassen C_k, einteilen lässt:

1. **Excellent.** Der bestmögliche Zustand einer Strecke ist charakterisiert durch einen stabilen *routing delay* und trägt den Namen *excellent*.

2. **Fair.** Eine leichte Verschlechterung der Performance einer Strecke deutet sich durch eine wachsende Varianz des *routing delay* und einzelne statistische Ausreißer an. Dieser Zustand wird als *fair* bezeichnet.

3. **Poor.** Der Zustand *poor* repräsentiert leichte Überlast auf einer Strecke und ist charakterisiert durch eine wiederum größere Streuung der Messwerte.

4. **Bad.** Der verbleibende, schlechteste Zustand einer Strecke steht für starke Überlast des Netzwerks und wird als *bad* bezeichnet.

Das durchschnittliche Verhalten dieser Klassen lässt sich am besten durch Rayleigh-Verteilungen beschreiben, da Gamma-Verteilungen und Gamma-Mischverteilungen zu individuell für Performance-Klassen sind. Dadurch ist sichergestellt, dass diese die für jede beliebige Strecke im X-WiN anwendbar sind. Die dabei für die einzelnen Performance-Klassen bestimmten Rayleigh-Parameter wurden in Tabelle 1 zusammengefasst.

Die Aufgabe der Performance-Klassifizierung ist es nun, herauszufinden, welche der obigen Klassen für die Erzeugung der Messwerte des *routing delay* eines bestimmten Zeitintervalls am ehesten in Frage kommt. Die Wahrscheinlichkeit, dass die beobachtete Abfolge $\mathbf{X}$ von der Performance-Klasse C_k generiert wurde, wird als $P(C_k|\mathbf{X})$ bezeichnet und lässt sich durch den Satz von Bayes bestimmen als:

$$P(C_k|\mathbf{X}) = \frac{\prod_{i=1}^{n} \mathcal{R}(x_i|\sigma_k^2)}{\sum_{m=1}^{4} \prod_{i=1}^{n} \mathcal{R}(x_i|\sigma_m^2)} \tag{5}$$

wobei es sich bei $\mathcal{R}(x|\sigma_k^2)$ die für Klasse C_k geschätzte Rayleigh-Verteilung handelt.

Nach der Berechnung der Wahrscheinlichkeiten $P(C_k|\mathbf{X})$ für alle Klassen wird das beobachtete Muster des *routing delay* schließlich derjenigen Performance-Klasse zugeordnet, für die $P(C_k|\mathbf{X})$ maximal ist, die also am wahrscheinlichsten für die Erzeugung von $\mathbf{X}$ verantwortlich ist. Dadurch lassen sich kritische Situationen wie Überlast auf einer Strecke leicht identifizieren.

Eine beispielhafte Anwendung der Performance-Klassifizierung für die Messwerte eines Tages ist in Abbildung 3 illustriert.

Excellent	Fair	Poor	Bad
932.4	1 492.6	210 511.7	20 492 767.0

Tabelle 1: Geschätzte Rayleigh-Parameter σ_k der Performance-Klassen

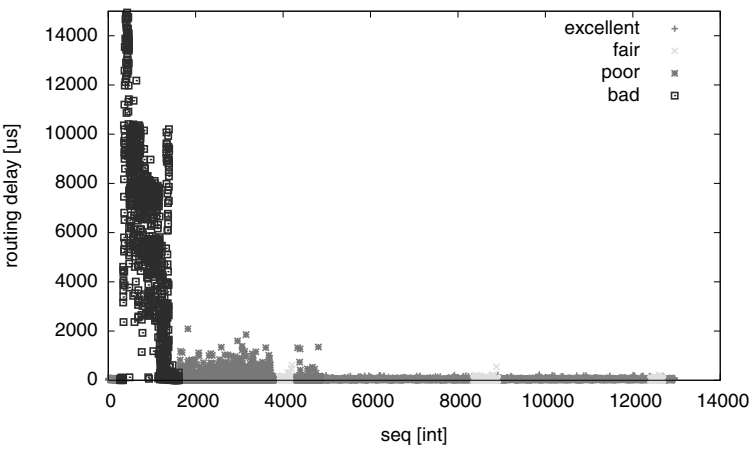


Abbildung 3: Ergebnisse der Performance-Klassifizierung für den am 21. Oktober 2007 gemessenen *routing delay* zwischen GÉANT2-Messboxen in Thessaloniki und Wien. Kurz nach Mitternacht wird die Performance dominiert durch starke Überlast. Erst gegen Morgen normalisiert sich der Zustand der Strecke wieder.

SRC	DST	#Excellent	#Fair	#Poor	#Bad	Score
A	B	3	4	20	21	37
A	C	7	0	29	12	58
...	...	...	...	...	...	...
K	P	35	13	0	0	131
C	F	41	6	1	0	136

Tabelle 2: Illustration eines Rankings abstrakter Strecken. Die Spalte #Excellent gibt die Anzahl der Zeitintervalle an, in denen die Performance-Klassifizierung für die entsprechende Strecke *excellent* ergab, #Fair die Anzahl für *fair* usw. Der Score wird schließlich als gewichtete Summe der Anzahlen berechnet, wobei der Faktor für *excellent* 3 ist, für *fair* 2, für *poor* 1 und für *bad* 0. Je geringer am Ende der Score einer Strecke ist, desto schlechter schneidet sie im Vergleich ab.

7 Ausblick

Auf Basis dieser Forschungsergebnisse entwickelt das WiN-Labor aktuell ein System, das in der Lage ist, einerseits im Falle kritischer Situationen Alarme zu generieren und auf der anderen Seite Langzeit-Auswertungen der Performance aller Strecken durchzuführen:

1. **Echtzeit-Alarmsystem.** Das Alarmsystem analysiert die neu generierten Messwerte in Form eines Datenstromsystems in Echtzeit und generiert in erkannten kritischen Netzwerksituationen wie z.B. Umrouting oder mittlerer bis starker Überlast Warnungen und Alarme.

2. **Langzeit-Analysesystem.** Darüber hinaus wird ein Analysetool entwickelt, das darauf ausgelegt ist, die durchschnittliche Performance aller verfügbaren Strecken im X-WiN über einen längeren Zeitraum zu bestimmen. In einem Ranking können dann schließlich die Strecken, wie in Tabelle 2 illustriert, miteinander verglichen werden. Dies ermöglicht u.a. die Identifikation der *low performers* unter ihnen, und liefert somit wichtige Informationen für Netzwerkspezialisten im Hinblick auf Schwachstellen im Netzwerk, die es zu beheben gilt.

Literatur

[AKZ99a] G. Almes, S. Kalidindi, and M. Zekauskas. A One-way Delay Metric for IPPM. `http://www.rfc-editor.org/rfc/rfc2679.txt`, September 1999. Online resource, accessed 2008-05-08.

[AKZ99b] G. Almes, S. Kalidindi, and M. Zekauskas. A One-way Packet Loss Metric for IPPM. `http://www.rfc-editor.org/rfc/rfc2680.txt`, September 1999. Online resource, accessed 2008-05-08.

[Bis06] C. Bishop. *Pattern Recognition and Machine Learning*. Springer, Berlin, 2006.

[CCM+04] B.-K. Choi, B.-K. Choi, S. Moon, Zhi-Li Zhang, K. Papagiannaki, and C. Diot. Analysis of point-to-point packet delay in an operational network. In S. Moon, editor, *Proceedings INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1797–1807, 2004.

[DC02] C. Demichelis and P. Chimento. IP Packet Delay Variation Metric for IP Performance Metrics (IPPM). `http://www.rfc-editor.org/rfc/rfc3393.txt`, November 2002. Online resource, accessed 2008-05-08.

[DLR77] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[EM99] T. Elteto and S. Molnar. On the distribution of round-trip times in TCP/IP networks. In *Proceedings of the 24th Conference on Local Computer Networks*, pages 172–181, 1999.

[FPP98] D. Freedman, R. Pisani, and R. Purves. *Statistics*. W.W. Norton & Company, New York, 3rd edition, 1998.

[GEA] The GÉANT2 Network. `http://www.geant2.net/`. Online resource, accessed 2008-05-08.

[HAD] The Hades measurement tool. Performance measurement in Internet Backbones. `http://www-win.rrze.uni-erlangen.de/ippm/messprogramm.html`. Online resource, accessed 2008-05-08.

[HC78] R. Hogg and A. Craig. *Introduction to Mathematical Statistics*. Macmillan, New York, 4th edition, 1978.

[HKK+06] P. Holleczek, R. Karch, R. Kleineisel, S. Kraft, J. Reinwand, and V. Venus. Statistical Characteristics of Active IP One Way Delay Measurements. In R. Karch, editor, *Proc. International Conference on Networking and Services ICNS '06*, pages 1–1, 2006.

[Hof01] G. Hofmann. Implementation eines Programms zur Bestimmung von Dienstgüte in IP-Netzen. Master's thesis, Friedrich-Alexander University of Erlangen-Nuremberg, 2001.

[Hol08] T. Holleczek. Statistical Analysis of IP Perfromance Metrics in International Research and Educational Networks. Master's thesis, Friedrich-Alexander University of Erlangen-Nuremberg, May 2008.

[HP06] J.A. Hernández and I.W. Phillips. Weibull mixture model to characterise end-to-end Internet delay at coarse time-scales. *IEE Proceedings-Communications*, 153(2):295–304, 2006.

[KR07] J.F. Kurose and K.W. Ross. *Computer Networking: A Top-Down Approach*. Addison-Wesley, 4th edition, 2007.

[Mac67] J.B. Macqueen. Some methods of classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[MC06] A. Morton and L. Ciavattone. Packet Reordering Metrics. `http://www.rfc-editor.org/rfc/rfc4737.txt`, November 2006. Online resource, accessed 2008-05-08.

[MP99] J. Mahdavi and V. Paxson. IPPM Metrics for Measuring Connectivity. http://www.rfc-editor.org/rfc/rfc2678.txt, September 1999. Online resource, accessed 2008-05-08.

[Muk94] A. Mukherjee. On the Dynamics and Significance of Low Frequency Components of Internet Load. *Internetworking: Research and Experience*, 5:163–205, 1994.

[NJ06] S. Naegele-Jackson. *Network-QoS and Quality Perception of Compressed and Uncompressed High-Resolution Video Transmissions*. PhD thesis, Friedrich-Alexander University of Erlangen-Nuremberg, 2006.

[NJKH04] S. Naegele-Jackson, R. Kleineisel, and P. Holleczek. IPPM Measurements and Network Load Behavior of the German Research Network G-WiN. In *Proceedings of the International Conference on Computing, Communications and Control Technologies (CCCT 04)*, pages 390–395, Austin, 2004.

[PAMM98] V. Paxson, G. Almes, J. Mahdavi, and M. Mathis. Framework for IP Performance Metrics. http://www.rfc-editor.org/rfc/rfc2330.txt, May 1998. Online resource, accessed 2008-05-08.

[Pap91] A. Papoulis. *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, New York, 3rd edition, 1991.

[PMF+03] K. Papagiannaki, S. Moon, C. Fraleigh, P. Thiran, and C. Diot. Measurement and analysis of single-hop delay on an IP backbone networks. *IEEE Journal on Selected Areas in Communications*, 21(6):908–921, 2003.

[Ric07] J.A. Rice. *Mathematical Statistics and Data Analysis*. Thomson, Belmont, 3rd edition, 2007.

[RN02] S. Russel and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, Upper Saddle River, 2nd edition, 2002.

[STK+06] S. Shalunov, B. Teitelbaum, A. Karp, J. Boote, and M. Zekauskas. A One-way Active Measurement Protocol (OWAMP). http://www.rfc-editor.org/rfc/rfc4656.txt, September 2006. Online resource, accessed 2008-05-08.

[Tan03] A.S. Tanenbaum. *Computer Networks*. Pearson Education International, 3rd edition, 2003.

[XWI] X-WiN – Germany's National Research and Educational Network. http://www.dfn.de/content/xwin/. Online resource, accessed 2008-05-08.

Interactive Analysis of NetFlows for Misuse Detection in Large IP Networks

Florian Mansmann, Fabian Fischer, Daniel A. Keim, Stephan Pietzko, Marcel Waldvogel

first.lastname@uni-konstanz.de

Abstract:
While more and more applications require higher network bandwidth, there is also a tendency that large portions of this bandwidth are misused for dubious purposes, such as unauthorized VoIP, file sharing, or criminal botnet activity. Automatic intrusion detection methods can detect a large portion of such misuse, but novel patterns can only be detected by humans. Moreover, interpretation of large amounts of alerts imposes new challenges on the analysts. The goal of this paper is to present the visual analysis system NFlowVis to interactively detect unwanted usage of the network infrastructure either by pivoting NetFlows using IDS alerts or by specifying usage patterns, such as sets of suspicious port numbers. Thereby, our work focuses on providing a scalable approach to store and retrieve large quantities of NetFlows by means of a database management system.

1 Introduction

Network administrators have a tendency to automate as many tasks as possible in order to keep pace with the ever increasing bandwidth requirements of modern network applications. Larger networks, in particular, have become unmanageable without smart intrusion detection systems. However, when it comes to analyzing attacks or detecting novel attacks, these systems only support the analyst in a very limited way.
While administrating our university network with several thousand hosts, we have realized that most of these systems generate a tremendous amount of alerts when being used in an open network setting with only few firewall restrictions as demanded by our users. In addition to that, it is hard to reason about the generated alerts since many of these systems are designed as blackboxes to guard the technological advance of the security provider.
In this paper we propose a novel system called *NFlowVis*, which is designed to visually present service usage and threats in the local IP network. Thereby, alerts from intrusion detection systems or defined application ports are used to identify potential attackers and visualize all their traffic to hosts within the administrated network in the next step. Note that the IP addresses in the figures are anonymized to protect the privacy of our users.
The rest of this paper is structured as follows: Section 2 presents related work, Section 3 discusses processing and querying challenges and solutions for large NetFlow data sets and Section 4 details our proposed visualization system. Afterwards, Section 5 shows how the tool is being used in practice. The last section concludes our work.

2 Related Work

Visualization for computer security is a relatively young research field. While substantial research has been conducted in the field in the last few years, for brevity this section will focus on visual network traffic monitoring and discuss the roots of the used visualization concepts. Please refer to the following books to gain deeper insight into the statistical, visualization, and application aspects of intrusion detection [Mar01, Con07, Mar08].

In the Open Source community, there are two popular tools: *NfSen* [NfS07] and *Stager* [Osl06]. Both tools comprise web frontends to display aggregated information about previously captured NetFlows. In the backend, database management systems enable efficient access to detailed information and efficient generation of aggregated reports. For visual analysis, both systems use line charts for displaying temporal overviews of network system load. While Stager only stores highly aggregated data, NfSen reverts back to the original flow data for detailed analysis.

Since network monitoring is particularly important for the health of the commercial network infrastructure, there exist a multitude of commercial systems. In contrast to the previously discussed tool, commercial systems such as *IBM Aurora*[1], *NetQoS Reporter Analyzer*[2], *Caligare Flow Inspector*[3], and *Arbor Peakflow*[4] often include methods for intrusion detection in which generated alerts can be examined through interactive reports. However, the used statistical charts and diagrams only scale to a limited number of alerts or highly aggregated information.

Visualization approaches in network monitoring aim at supporting the system administrator in the exploration of network traffic by means of interactive visual displays. *NVisionIP* [LBS+05], for example, enables visual pattern recognition and drill-down functionalities to inspect suspicious machines. *TNV* [GLRK05] is a network traffic visualization tool focusing on temporal aspects by means of a time versus internal host matrix, which details traffic flows for each host and links the external communication partners on the side. The home-centric network view of *VISUAL* [BFN04] is probably closest to our proposed visualization since a matrix showing all internal hosts in the center is linked to external communication partners using straight connecting lines.

In contrast to this work, we made two major conceptual changes: a) Instead of using a matrix view for the internal hosts, we employ a *TreeMap* [Shn92] visualization, which hierarchically maps the monitored network infrastructure to prefixes of various granularity. Unlike in our previous work [MKN+07], high-load entities are thereby enlarged. b) Rather than using straight lines to link the communication partners, we employ *Hierarchical Edge Bundles* [Hol06] to visually group related flows, and thereby avoid visual clutter. While we visualized flows using Hierarchical Edge Bundles with both start and end point within a TreeMap visualization in an earlier work [MFKN07], the work presented in this paper explicitly focuses on a home-centric network view, which represents the local IP prefixes or addresses in a TreeMap and places the external hosts at its border.

Abstract graph representations normally seek a way to effectively use the available screen

[1] `http://www.zurich.ibm.com/aurora`
[2] `http://netqos.com/solutions/reporteranalyzer`
[3] `http://www.caligare.com/netflow`
[4] `http://www.arbornetworks.com`

space. Thereby, linked nodes are rendered close to each other to avoid visual clutter caused by crossing edges. Cheswick et al., for example, mapped a graph of about 88 000 networks as nodes having more than 100 000 connecting edges [CBB00], obtained by measuring the quality of network connections in the Internet from different vantage points.
The study in [TN+00] goes one step further in the automated analysis by applying clustering methods on graph structures, in order to reveal similar attack structures.
There exist hybrid approaches that partly take geographic information into account while calculating the graph layout on the screen. One such approach is the visualization interface of the *Skitter* application that uses polar coordinates to visualize the Internet infrastructure [Cla01]. Each AS node's polar coordinate is determined by the geographical longitude of its headquarter and by the hierarchical connectivity information.
The implementation of the node link diagrams in our tool can rather be seen as a feature than as a novel research contribution since we only apply efficient graph layout and interaction frameworks.

3 Large-Scale Processing of NetFlows

A big challenge in the analysis of network related records is the great amount of data to be handled in real-time, especially when trying to avoid packet loss. To use the proposed analysis system we are required to store all available NetFlow information in a relational database management system. This means to cope with three main problems.
Firstly, we need to receive NetFlow streams in real-time. We need to accept these streams immediately and on link speed, which are later processed in 5 minutes intervals. The underlying protocol utilizing the NetFlow streams is the stateless UDP protocol. The system receiving the streams has to be as fast as possible to accept all records even in peak times. To accomplish these requirements we set up a NetFlow collector server using a flow capture daemon (flow-tools [RFL00]) and storing the incoming NetFlow streams directly to the RAM of the server system to prevent a possible I/O bottleneck at this early stage. Having a few GB of RAM storage we can use this memory as a buffer cache to prevent packet loss and to provide more time for the next more time-consuming preprocessing and analysis steps.
Secondly, we have to preprocess the incoming NetFlow streams and to transform them to a format which can be imported to our database in a fast and efficient way. We were required to use batch import functionalities, so a very convenient data format are plain comma separated text files. In this step we also integrated an anonymization filter to use the system for scientific purposes and to prevent scientists to access unanonymized data. To overcome privacy concerns we integrated the anonymization process (especially in the testing phase) at this early stage, to ensure all data available in the database is completely anonymized. In production use it is easy for the network security officer to disable the anonymization process, which will also lead in a higher overall performance of the system. Note that this is not an inherent function of the system for operational use.
The third step of the processing workflow is to actually import the available data to the database system and to store it in a scheme which makes the data analyzable by NFlowVis.

This step is not done on the fly like the previous steps. The server system will automatically import the data to the database server in regular intervals based on the required analysis timeframe. Because of the limitations of our hardware we had to restrict the import interval to one day for now, but we are adapting our approach to an import interval to few hours using a faster database server with more memory.
The main bottleneck of the processing system is the index creation during the batch import of the data. This can be improved by importing the records to empty tables without pre-existing indices. To have a reasonable performance and response time querying the database, indices are still required. By creating the indices after the batch import of the new records is finished, we were able to drastically increase the import time. To support this importing scheme directly through the underlying database layout, we introduced a chronical timeframe table hierarchy, in which each table presents one import interval (e.g. one day). Through the heritage structure or through joining, it is basically still possible to access full years, months or several days. Additionally the system automatically creates several pre-aggregated tables during the import process to further improve the querying performance and support specific queries used in NFlowVis to visualize the data.

4 Visual Analysis of NetFlows

Keeping the general workflow of a network analyst in mind, we developed *NFlowVis* to interpret the relevance of network security alerts. The system supports this full workflow through its five analysis views with a general network *overview*, an integrated *intrusion detection view*, the *flow visualization* of attackers' connections, a detailed *host view*, and the full *NetFlow records* of the specified communications as the most detailed view. In the graphical user interface these views are represented through several tabs to emphasize the drill-down and filtering process. Figure 1 describes the design of NFlowVis, showing project selection (A), key data of the selected project (B), the Quick Lookup interface for directly querying specific IP addresses (C), fast access to external tools (D) and the data exploration views ordered according to the levels of details (E). Within the *overview* tab, the system provides several user-defined plots (F). With the help of these graphs the analyst is able to get a rough overview of the actual network situation and utilization detailing the aggregated traffic and port usage within the whole network. To visualize these time series we use line charts and grouped line-wise pixel arrangements. The use of both visualizations combines the advantages of the well known line charts and pixel visualization, which provides identification of every single minute and enables recognition of recurring patterns. The overview also provides an interactive port activity map to identify the most active ports.
The *intrusion detection view* in Figure 2 is key to system since it links our NetFlow exploration system with an intrusion detection system by showing the imported alerts. Note that the only requirement for this table is that the first row contains the IP address of an attacker, whereas the number of additional columns are only relevant for the human analyst. For further investigation of a number of attacking hosts, it is possible to select the attackers and to visualize their traffic with hosts in our network to explore their influence.

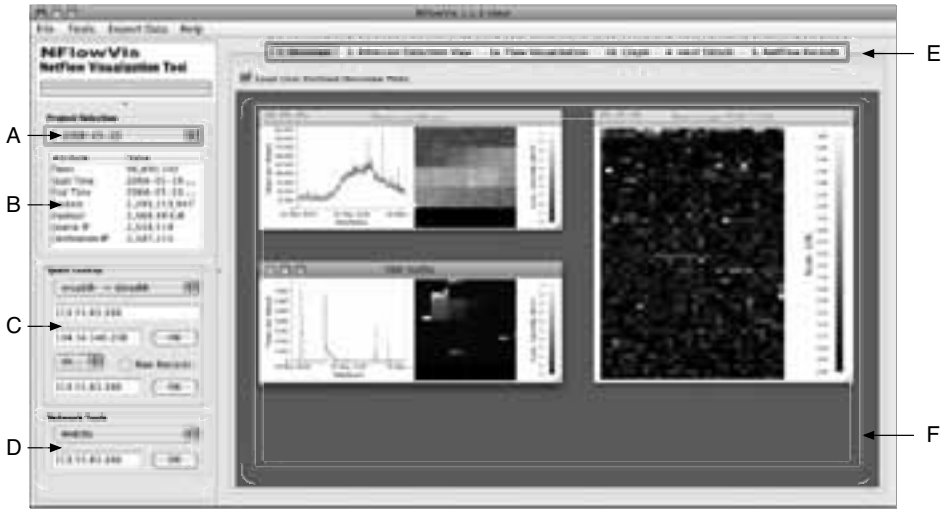


Figure 1: User interface of the NFlowVis system showing the annotated main view. In this start display the user can choose a dataset (A), see some overall statistics of this data set (B), directly access detailed data for a particular host or host combination (C), use external tools to query background information on a host (D), access a few user-defined plots (F) showing aggregated flows per minute (top left), traffic on a particular port (bottom right) or the activity on the most used ports (right) or start a detailed analysis (E).

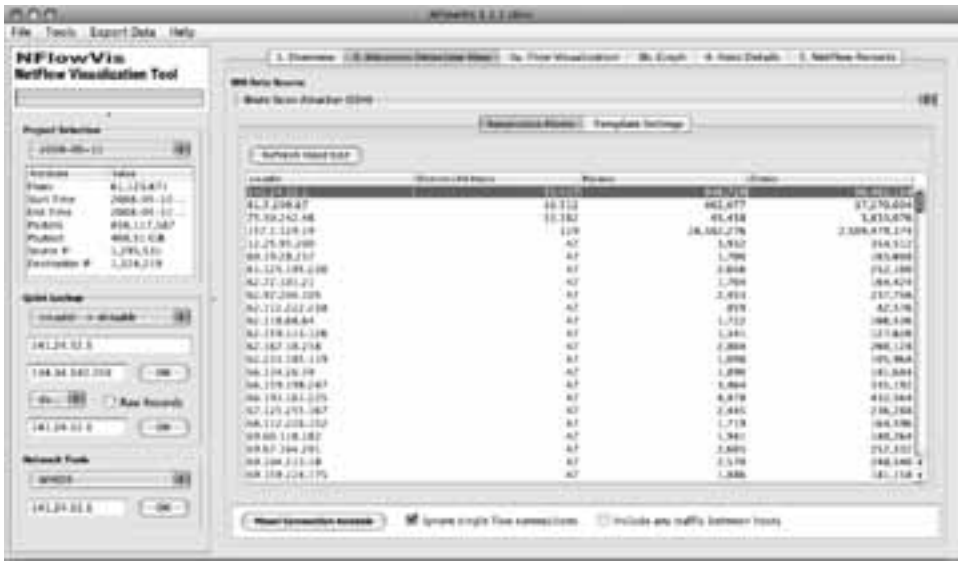


Figure 2: Alerts originating from an external IDS or warning lists in the Intrusion Detection View

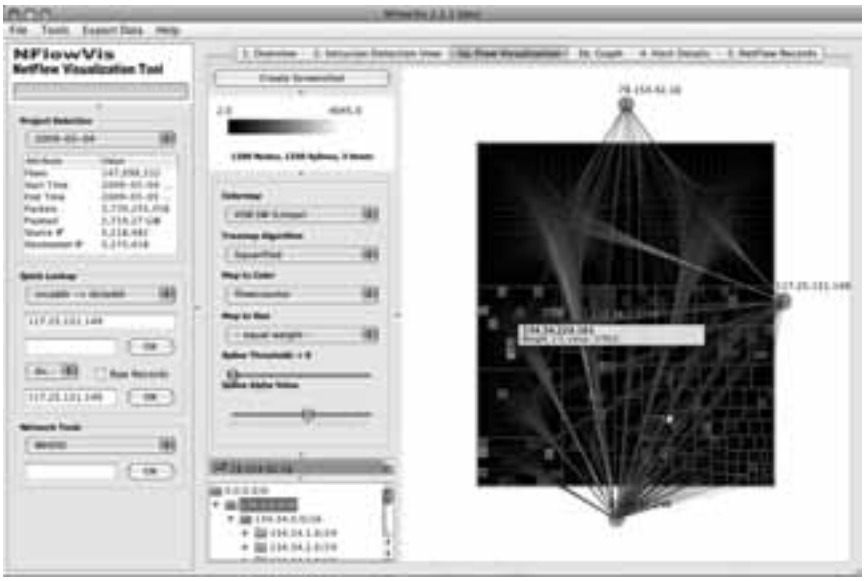


Figure 3: Flow Visualization of traffic patterns displaying the internal network structure as a TreeMap and the external hosts on the borders. The internal hosts are displayed as rectangles, which are contained within their upper level prefixes. Their size and color are configurable to the traffic payload, the number of flows or packets, or to a constant (e.g, equal-size) using the configuration on the left. The visualization shows three external hosts scanning for open SSH ports; the upper two prefixes contain a lot of scanned hosts, but the number of flows is always low (black color). In contrary to this, the prefixes in the lower part contain less scanned hosts but some hosts received a lot of flows. The yellow host, for example, received 1770 flows from the three attackers.

Besides the integration of external IDS alerts and warning lists, this view also provides a template editor to define database queries, which can directly access arbitrary tables. We included a variety of different predefined warning lists, such as grabbing all SSH traffic or other suspicious activities.

Within the *flow visualization* view shown in Figure 3, we map the monitored network to a TreeMap visualization in the center of the display and arrange the previously selected attackers at the borders. The TreeMap comprises all hosts related to the attacking hosts during the chosen timeframe, which can be defined in the project creation wizard. Flows between the attackers and the local hosts or prefixes are displayed through Splines, whose control points are the center points of the network prefixes of various levels and the attackers on the outside. The size of the TreeMap rectangles (weight) and their background color can be set to arbitrary attributes of the aggregated flow data, e.g., flow count, transferred packets, or bytes. Furthermore, splines representing traffic links smaller than a selected threshold can be discarded or made less visible by adjusting the sliders on the configuration panel on the left.

In the default configuration the Spline color correlates with the attacker's IP prefix, which better shows the behavior of attackers with similar prefixes supporting the analyst in gaining insight into the distribution of the attacking IP addresses.

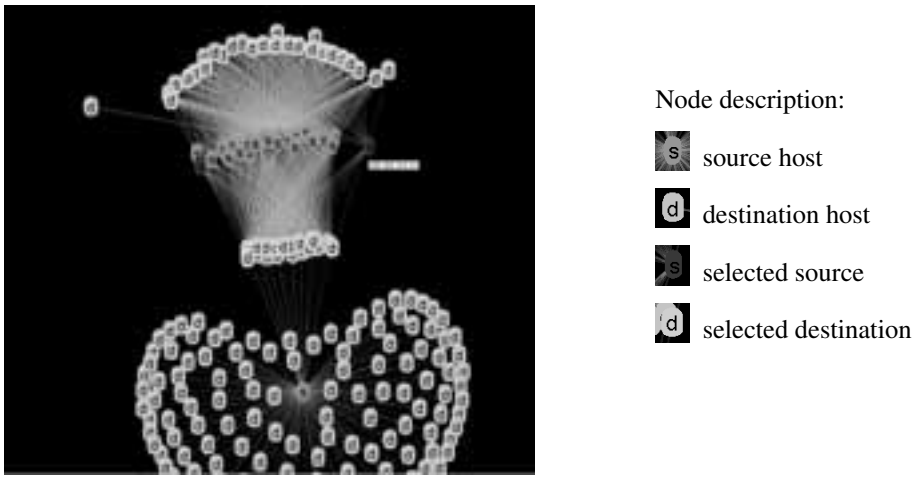


Figure 4: Graph view showing a network scan (below) and an attack from a botnet (above).

The position of the attackers is calculated based on a *k-Medoid* clustering algorithm [KR90], which identifies all attackers and clusters them based on similar destination hosts. Therefore, it is possible to arrange hosts with similar victims close to each other to minimize overlaps. Another positive effect is the meaningful grouping of collaborating attackers in the same cluster.

Figure 4 shows the graph view, which can be seen as an alternative to the previously presented flow visualization. By extracting the communicating hosts from the traffic specified in the IDS view, we generate a node link diagram using the GraphViz tool [EGK+02] to efficiently calculate the layout and the Prefuse toolkit [HCL05] for displaying and interacting with the nodes. Note that the choice between using this graph layout or the previously introduced home-centric TreeMap visualization depends on the analysis task. While the graph view better presents the structure between the attackers and its victims, the home-centric visualization helps to identify properties of the attack that are influenced by the local network infrastructure. A pool of computers running an unpatched operation system, for example, could be easily identified in the home-centric network visualization due to the rectangle grouping, whereas extracting this information from the graph layout would involve interactively displaying one IP address after the other.

For further analysis of single hosts under attack, the analyst is able to select hosts in the two latter visualizations, which triggers the *host view* detailing histograms, a port activity map which visualizes the data volume on the used port numbers, and an aggregated overview of all attackers related to the chosen host (see Figure 5). Likewise, the original NetFlow records can be further analyzed by drilling-down and extracting the corresponding data in the *NetFlow records* view showing the timestamp, source/destination hosts and prots, the protocol as well as the number of data packets and octets aggregated on flow level.

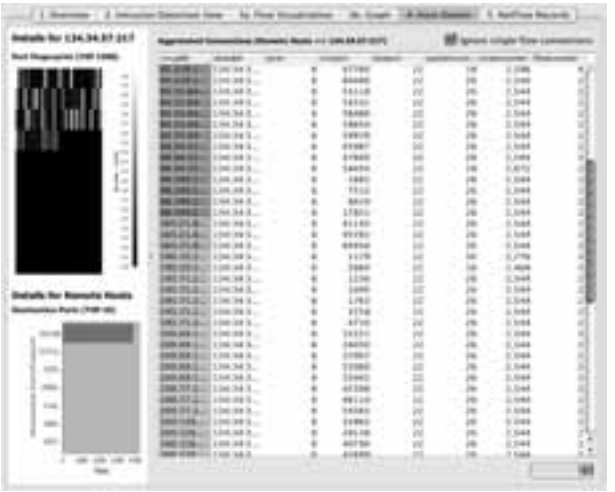


Figure 5: Host details view

5 Results and Findings

While applying the tool for monitoring our university network at the gateway, we found several interesting patterns. The first pattern that usually sticks out is caused by network scans, such as the scan for open remote desktop ports in Figure 3 or the scan for open VNC servers in Figure 5. These patterns were detected after specifying the respective ports in an SQL database query in the IDS view. While these scans can automatically be detected by relatively simple detection algorithms, the visualization can reveal further details of the structure of the attack and give additional indications whether the attack was successful or not. This is done by specifying that all traffic between the external attacker and the internal victim host is visualized. After having found a valid user and password combination, the attacker usually logs into the system and downloads a malware application to control the conquered host. While unsuccessful attacks often result in relatively little traffic, successful attacks might result in additional traffic on other ports. We show traffic properties for each internal host in its rectangle size and color to guide the analyst to these suspicious hosts, which have a higher probability of being hacked.

We were furthermore able to identify botnet attacks on open SSH ports as displayed in Figure 5. The used clustering algorithm thereby groups external hosts, which connect to similar sets of internal hosts, on the borders, thereby resulting in a more insightful visualization. Note that while this flow visualization focuses on an internal view of the network, which uses prefix information to group subnets, the graph view shown in Figure 4 might give additional hints to structures of attacks.

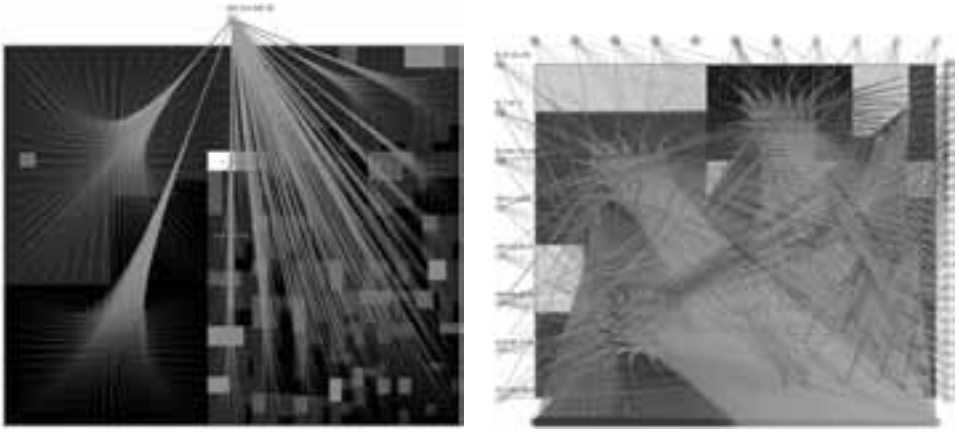


Figure 6: Application examples: Scan for Virtual Network Servers using the VNC protocol on December 8, 2008 (left) and SSH attack from a botnet on November 29, 2008 (right)

6 Conclusions

In this paper we presented the NFlowVis system for analyzing large amounts of NetFlows and intrusion detection alerts. In contrast to traditional IDS, we pursued a visual data analysis approach since this allows the experts to gain deeper insight into current threat situation and to discover novel attacks. In particular, we presented two complementary visualization approaches for the analysis of attacker and victim hosts. The first approach is comprised of a local network centric TreeMap view, which groups local network hosts according to their prefix information and allows the analyst to draw conclusions about the focus areas of attacks within the network. The second approach uses methods from graph drawing to visualize the link information between the attackers and their victims and can be especially helpful to distinguish between distributed scans and attacks.
For future work, we plan to create a database independent application, which allows administrators to analyze smaller tcpdump/NetFlow files without using a database server.
This work has been funded as part of the BW-FIT research cluster “Gigapixel displays” by the German federal state Baden-Württemberg.

Bibliography

[BFN04] R. Ball, G.A. Fink, and C. North. Home-centric visualization of network traffic for security administration. *Proceedings of the 2004 ACM workshop on Visualization and data mining for computer security*, pages 55–64, 2004.

[CBB00] Bill Cheswick, H. Burch, and S. Branigan. Mapping and Visualizing the Internet. In *Proceedings of the USENIX Annual Techincal Conference*, 2000.

[Cla01] K.C. Claffy. CAIDA: Visualizing the Internet. *IEEE Internet Computing*, 05(1), 2001.

[Con07] Greg Conti. *Security Data Visualization - Graphical Techniques for Network Analysis*. No Starch Press, 2007.

[EGK+02] J. Ellson, E. Gansner, L. Koutsofios, S.C. North, and G. Woodhull. Graphviz-Open Source Graph Drawing Tools. *LECTURE NOTES IN COMPUTER SCIENCE*, pages 483–484, 2002.

[GLRK05] John R. Goodall, Wayne G. Lutters, Penny Rheingans, and Anita Komlodi. Preserving the Big Picture: Visual Network Traffic Analysis with TNV. In *VIZSEC '05: Proceedings of the IEEE Workshops on Visualization for Computer Security*, Washington, DC, USA, 2005. IEEE Computer Society.

[HCL05] J. Heer, S.K. Card, and J.A. Landay. prefuse: a toolkit for interactive information visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 421–430. ACM New York, NY, USA, 2005.

[Hol06] Danny Holten. Hierarchical Edge Bundles: Visualization of Adjacency Relations in Hierarchical Data. *IEEE Trans. Vis. Comput. Graph.*, 12(5):741–748, 2006.

[KR90] L. Kaufman and P.J. Rousseeuw. Finding groups in data. An introduction to cluster analysis. *Wiley Series in Probability and Mathematical Statistics. Applied Probability and Statistics, New York: Wiley*, 1990.

[LBS+05] K. Lakkaraju, R. Bearavolu, A. Slagell, W. Yurcik, and S. North. Closing-the-Loop in NVisionIP: Integrating Discovery and Search in Security Visualizations. In *Visualization for Computer Security, IEEE Workshops on*, pages 9–9, 26 Oct. 2005.

[Mar01] David J. Marchette. *Computer Intrusion Detection and Network Monitoring: A Statistical Viewpoint*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2001.

[Mar08] Raffael Marty. *Applied Security Visualization*. Addison-Wesley Professional, 2008.

[MFKN07] F. Mansmann, F. Fischer, D. Keim, and S. North. Visualizing large-scale IP traffic flows. In *Proceedings of 12th International Workshop Vision, Modeling, and Visualization*, 2007.

[MKN+07] Florian Mansmann, Daniel A. Keim, Stephen C. North, Brian Rexroad, and Daniel Sheleheda. Visual Analysis of Network Traffic for Resource Planning, Interactive Monitoring, and Interpretation of Security Threats. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1105–1112, 2007.

[NfS07] NfSen - Netflow Sensor. A graphical web based front end for the nfdump netflow tools, 2007. `http://nfsen.sourceforge.net/`.

[Osl06] A. Oslebo. Stager A Web Based Application for Presenting Network Statistics. In *Network Operations and Management Symposium, 2006. NOMS 2006. 10th IEEE/IFIP*, pages 1–15, 2006.

[RFL00] Steve Romig, Mark Fullmer, and Ron Luman. The OSU Flow-tools Package and CISCO NetFlow Logs. In *LISA '00: Proceedings of the 14th USENIX conference on System administration*, pages 291–304, Berkeley, CA, USA, 2000. USENIX Association. `http://www.splintered.net/sw/flow-tools/`.

[Shn92] Ben Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Trans. Graph.*, 11(1):92–99, 1992.

[TN+00] J. Toelle, O. Niggemann, et al. Supporting intrusion detection by graph clustering and graph drawing. In *Proceedings of Third International Workshop on Recent Advances in Intrusion Detection RAID*, 2000.

Messen und Schalten im Rechenzentrum: Kostengünstige Sensorknoten mit sicherer Anbindung an offene Netze

Michel Steichen, Dirk Henrici, Paul Müller
{m_steich, henrici, pmueller}@informatik.uni-kl.de

Abstract: Die in Rechenzentren eingesetzten vernetzten Sensorknoten zum Messen und Schalten sind heute in den meisten Fällen unabhängige und kostspielige Endgeräte. Intention dieses Beitrages ist es, ein verbessertes Konzept für solche Sensorknoten vorzustellen. Dabei handelt es sich keinesfalls um eine Nachimplementierung bestehender Lösungen. Vielmehr wurden in dieser Arbeit die Schwächen aktueller Lösungen und Lösungsvorschläge analysiert und darauf basierend eine neue, generische Architektur entworfen und prototypisch implementiert, die auch in anderen Bereichen, wie z.B. dem Assisted Living, einsetzbar ist. Berücksichtigte Anforderungen sind unter anderem Anschaffungskosten, Energieverbrauch im Betrieb, Benutzerfreundlichkeit, Skalierbarkeit, Sicherheit, Zuverlässigkeit und Flexibilität. Um all diese Anforderungen erfüllen zu können, wurden die Aufgaben der Sensorknoten in ein Modell aus mehreren Schichten aufgeteilt.

1 Einleitung

Mit Hilfe von Sensorknoten lassen sich Umgebungszustände überwachen und Aktuatoren steuern. Die Technologie ist zur Gebäudeautomation und damit auch zur Überwachung von Rechenzentren geeignet. Die Aufgaben dieser Geräte sind vielfältig. Passive Datenerfassung, wie beispielsweise die Messung von Temperatur, Luftfeuchtigkeit und Stromverbrauch ermöglichen eine gezielte und schnelle Zustandserfassung der aktuellen Gegebenheiten. Mittels aktiver Komponenten, wie Relais, anderen Aktuatoren oder Displays, besteht die Möglichkeit mit der Umgebung zu interagieren.

In dieser Arbeit werden Sensorknoten vorgestellt, die auf einer flexiblen Schichtenarchitektur [1] basieren. Im Folgenden werden die Gründe und die Absichten erläutert, die die Neuentwicklung solcher Sensorknoten motivieren.

1.1 Motivation und Ziel

Beim genaueren Betrachten der zahlreichen angebotenen Lösungen erkennt man in den meisten Systemen eine Reihe von Schwächen. Aus diesen Erkenntnissen heraus wurden die Anforderungen an das neue System identifiziert.

Ziel der Arbeit war es, ein neues Design für Sensorknoten zu entwickeln, welches Verbes-

serungen in den folgenden Bereichen aufzeigt:

1. Kosten in der Herstellung / im Erwerb
2. Energieverbrauch im Betrieb
3. Benutzerfreundlichkeit bei der Installation und der Konfiguration
4. Skalierbarkeit des Systems
5. Sicherheit beim Zugriff und der Datenübertragung
6. Zuverlässigkeit während des Betriebs

Um sämtlichen Anforderungen gerecht zu werden, war es notwendig, die Funktionalität von Sensorknoten in ein Modell aus mehreren Schichten aufzuteilen. Eine solche Vorgehensweise ermöglicht es, die "Intelligenz" und Komplexität auf der untersten Ebene auf ein Minimum zu reduzieren und die Gesamtaufgabe zweckmäßig auf die verfügbaren Schichten aufzuteilen. Aufbau und Funktion der einzelnen Ebenen werden im dritten Abschnitt behandelt.

1.2 Problemstellung

Der Aufwand bei der Entwicklung von Sensorknoten teilt sich auf die drei Bereiche Hardware, Vernetzung und Software auf.

Hardware: Im Vordergrund steht hier die geeignete Wahl der Bauteile die, unter Berücksichtigung der Anforderungen, kostengünstig und energiesparend sein sollen.

Vernetzung: Für die Verbindung von Sensorknoten und Steuerknoten werden Bussysteme oder andere Datennetztechnologien benötigt. Neben der Übertragungsgeschwindigkeit stehen auch hier die Merkmale Energiebedarf und Kosten an vorderster Stelle. Beispielsweise benötigt Ethernet mit ca. 1 Watt pro Port deutlich mehr Energie als Feldbussysteme wie ein CAN-Bus oder Profibus. Darüber hinaus gibt es weitere Kriterien, etwa Quality-of-Service-Eigenschaften.

Software: Der Bereich Software beschäftigt sich mit der Implementierung von Protokollen, Sicherheitsalgorithmen, Dienstprogrammen zu Installations- und Konfigurationszwecken, sowie einer benutzerfreundlichen Steuer- und Benutzungssoftware. Die Firmware der Sensorknoten vermittelt zwischen Sensorhardware, Mikrocontroller und Schnittstellen.

1.3 Anforderungen an die Architektur

Im Allgemeinen sind Sensorknoten darauf ausgelegt, unterschiedliche physikalische Größen, beispielsweise Temperatur, Druck oder Helligkeit, in einen entsprechenden elektrischen Spannungswert oder eine entsprechende Stromstärke umzuwandeln. Diese analoge Eingabe wird dann mit Hilfe von A/D-Wandlern in einen digitalen Wert überführt. Neben analogen Werten gilt es aber auch binäre Zustände zu erfassen.

Neben der Datenerfassung spielt das Interagieren mit der Umwelt eine elementare Rolle. Die Hardware der Sensorknoten muss somit die Funktionalität besitzen, auch aktiv in die Umgebung einzugreifen zu können. In den meisten Fällen begrenzt sich dieses Steuern und Regeln auf das Ein- und Ausschalten von Verbrauchern. In Fällen, wo beispielsweise ein analoger Spannungswert gebraucht wird, können entsprechende D/A-Wandler eingesetzt werden. Displays sollten zur übersichtlichen Anzeige von Betriebszuständen ebenfalls angesteuert werden können.

Ein weiteres relevantes Merkmal der Architektur ist die benutzte Vernetzungstechnologie. Wie schon kurz angedeutet, wird das Sensornetz in diesem Beitrag auf mehrere Schichten aufgeteilt. Um den Anforderungen gerecht zu werden, ist die Hardware auf den untersten Schichten kostengünstig und ressourcensparend gehalten. Hier kommen Feldbusse zum Einsatz, auf höheren Schichten hingegen Ethernet. Im Rahmen der für diesen Beitrag durchgeführten Implementierung werden CAN-Busse als Feldbusse verwendet. Jedoch können prinzipiell zahlreiche unterschiedliche Bussysteme und auch drahtlose Übertragungstechniken zum Einsatz kommen.

Was die Eingaben angeht, so sollten die Sensorknoten Wert- und Zustandsänderungen eigenständig weitergeben. Das Sensornetz kann somit seine Umgebung aktiv überwachen. Ein permanentes Abfragen der aktuellen Werte und Zustände, sogenanntes “polling”, wird überflüssig. Besonders interessant ist die Frage, wann und wie oft Änderungen weiter geschickt werden. Bei binären Eingängen ist das Versenden einer Nachricht bei jedem Flankenwechsel ausreichend. Optional kann dazu noch die Zeit gemessen werden, die zwischen zwei Zustandsänderungen verstrichen ist. Die Reaktion auf analoge Eingabewerte erweist sich als komplizierter. Allgegenwärtiges Rauschen erzeugt kontinuierliche Schwankungen bei den Messergebnissen. Diese unerwünschten und chaotischen Änderungen können durch Tiefpässe geglättet werden. Falls dies nicht anwendbar ist, besteht die Möglichkeit, nur in periodischen Zeitabständen eine Wertänderung weiterzuleiten.

2 Aktuelle Lösungen und Stand der Technik

Es gibt eine Vielzahl von Konzepten und Produkten auf dem Markt, die auf unterschiedliche Anwendungsbereiche ausgerichtet sind und die oben identifizierten Anforderungen in unterschiedlichem Maße erfüllen. Im Folgenden sind einige Beispiele dargestellt, um das verfügbare Spektrum aufzuzeigen.

SCADA
SCADA [2] steht für “Supervisory Control and Data Acquisition”. Dabei handelt es sich um ein reines Konzept, welches der Überwachung und Steuerung von Industrieanlagen dient. Anwendungen von SCADA liegen im Bereich der Regelung unter Berücksichtigung von Echtzeitfähigkeit. Die Technologie ist in mehrere Schichten aufgeteilt. Die Sensoren und Aktuatoren werden von Fernbedienungsterminals (RTU) oder speicherprogrammierbaren Steuerungen (SPS) kontrolliert. Mit Hilfe einer Benutzerschnittstelle in den oberen Schichten können die erfassten Daten visualisiert werden, sowie das Verhalten des Systems gesteuert werden.

Da es sich bei SCADA um ein Konzept handelt und keine abgeschlossene Technologie, gibt es eine Reihe von Herstellern, die ihre eigenen proprietären Interpretationen von SCADA anbieten. Oft kommen hier schon auf unterster Ebene komplexe Techniken wie Ethernet und TCP/IP zum Einsatz.

LCN
LCN [3] (Local Control Network) ist ein proprietäres und universelles Gebäudeleitsystem. Die einzelnen Sensorknoten haben frei benutzbare Ein- und Ausgänge. Die Kommunikation der Knoten erfolgt über ein Bussystem. Ein bestehendes Netz lässt sich jederzeit leicht erweitern und mit Hilfe einer PC-Software steuern.

Nachteil dieser Technologie ist die erforderliche Verdrahtung und Installation durch einen qualifizierten Elektroinstallateur. Die Unterstützung von drahtlosen Komponenten würde das Nachrüsten in alten Gebäuden und Häusern wesentlich vereinfachen.

BTNode
Bei BTNode [4] handelt es sich um eine reine drahtlose Sensornetz-Plattform. Die von der ETH Zürich entwickelten Sensorknoten können autonome Ad-Hoc-Netzwerke untereinander aufbauen. Die Schnittstellen der Ein-und Ausgänge sind sehr generisch gehalten, und die Kommunikation erfolgt über Bluetooth.

Trotz der interessanten und leistungsfähigen Funktionen, die diese Knoten besitzen, erfüllen sie in mehreren Punkte keinesfalls die in diesem Beitrag angestrebten Anforderungen. BTNodes bestehen aus kostspieliger Hardware und sind leistungsfähige Endgeräte mit anderer Zielsetzung.

Im nächsten Kapitel folgt ein Einblick in die praktische Umsetzung eines Sensornetzes, welches die vorgestellten Anforderungen erfüllt.

3 Umsetzung der Anforderungen

Basierend auf den Designentscheidungen wird in diesem Abschnitt das Konzept der Schichtenarchitektur erklärt. Anschließend werden die Anforderungen in Bezug auf die Umsetzung auf das Schichtenmodell nochmals detaillierter formuliert.

3.1 Die Schichtenarchitektur

Auf den ersten Blick scheint ein Modell aus mehreren Schichten die Entwicklung eines Sensorknotensystems unnötig komplizierter zu machen. Doch durch eine solche Struktur vermeidet man genau die Probleme, die bei vielen Lösungen immer wieder auftauchen. Abbildung 1 gibt einen ersten Überblick über dieses Schichtenmodell.

In den Endgeräten, d.h. den Sensorknoten selbst, sollte möglichst wenig Funktionalität liegen, damit sie einfach und kostengünstig sind. Die “Intelligenz” des Systems wird auf

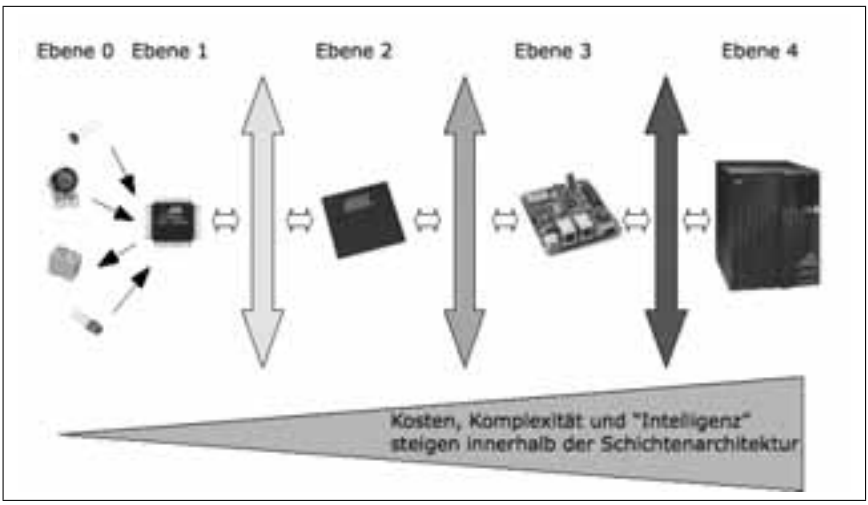


Abbildung 1: Schichtenarchitektur

die Hardware und Software der einzelnen Schichten aufgeteilt. Unter Berücksichtigung der Kosten und des Energiebedarfs gibt es ein Gefälle von den höheren Schichten aus zu den niedrigeren Schichten hin. Die unteren Ebenen enthalten jeweils nur soviel “Intelligenz”, wie unbedingt erforderlich ist. Somit befinden sich auf den Ebenen null und eins auch die günstigsten Komponenten des Sensornetzes. Ebene eins hat nur die Aufgabe, die Sensoren und Aktuatoren zu bedienen. Die gesammelten Information werden dabei so schnell wie möglich an die nächste Schicht weitergegeben.

Dieser Ansatz weist einige Vorteile auf. Soll eine Erweiterung des Netzes vorgenommen werden, so reicht es aus, neue Komponenten in den ersten beiden Schichten einzufügen. Die neu eingefügten Komponenten gliedern sich problemlos in das bestehende Netz ein und können von den darüber liegenden Schichten ohne großen Aufwand zusätzlich bedient werden. Damit erreicht man nicht nur eine Verringerung der Anschaffungskosten sondern zentralisiert und vereinfacht die Verwaltung und verringert die benötigte Hardware und die Komplexität.

Dies lässt sich am besten anhand eines Beispiels erläutern. In einem installierten Sensorknotennetz wird ein zusätzlicher Temperatursensor benötigt. Bei vielen konventionellen Sensorknoten geschieht eine solche Erweiterung durch den Kauf eines neuen Sensorknotens. Da es sich hierbei um komplexe selbstständige Endgeräte handelt, erwirbt der Kunde zusätzlich zum Sensor ein komplettes eingebettetes System, welches beispielsweise über einen Ethernet-Port und die Rechenleistung eines kleinen Webservers verfügt. Durch diesen zusätzlichen, nicht erforderlichen Ballast erhöhen sich die Kosten deutlich. Hier sind Anschaffungskosten der Komponente, Energieverbrauch und Kosten für benötigte Switchports zusammen zu rechnen. Um Denial-of-Service-Angriffe zu verhindern, müssen die Geräte vom Internet abgetrennt betrieben werden. Die Komplexität der Software kann aufwändige Updates erfordern.

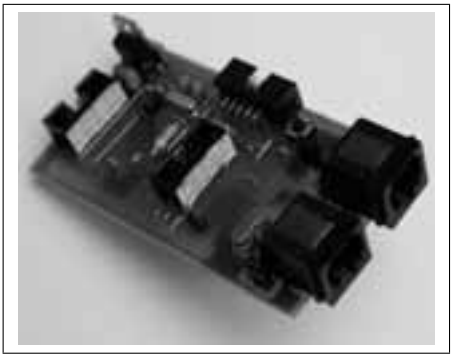


Abbildung 2: Sensorknoten

Durch das Schichtenmodell vermeidet man diese Kosten und Probleme. Der Bedarf eines zusätzlichen Temperatursensors reduziert sich auf die Anschaffung eines Sensorknotens auf der ersten Ebene. Dieser lässt sich benutzerfreundlich in das bestehende Netz einfügen. Die Kosten reduzieren sich auf ein Minimum, ebenso der Energiebedarf. Der neue Sensorknoten teilt sich nun, zusammen mit den Knoten seiner Ebene, das Rechenpotential der höheren Schichten, die kompliziertere Aufgaben bewältigen können. Somit fällt diese Lösung viel billiger und umweltfreundlicher aus.

Im folgenden Abschnitt wird die praktische Umsetzung der einzelnen Schichten in Bezug auf die identifizierten Anforderungen dargestellt.

3.2 Die Realisierung eines Prototypen

Im Rahmen einer Diplomarbeit [5] an der Technischen Universität Kaiserslautern wurde das geschilderte Schichtenmodell detailliert ausgearbeitet. Unter Berücksichtigung der Anforderungen wurde ein Prototyp entwickelt, der nun zur Klimaüberwachung im Rechenzentrum eingesetzt wird.

Sensoren/Aktuatoren und Sensorknoten: Ebenen null und eins
Die Sensorknoten, ein Beispiel ist in Abbildung 2 dargestellt, sind sowohl aus Sicht der Herstellung als auch der benötigten Betriebsenergie kosteneffizient. Bei der Auswahl der Mikroprozessoren und Controller wurde auf einen niedrigen Anschaffungspreis, eine gute Verfügbarkeit sowie eine stromsparende Ausführung geachtet. Die Wahl fiel dabei auf den 8-Bit Mikrocontroller ATmega8 von Atmel[1]. Die Hardwareressourcen des ATmegas sind mit 8 KB Flash-Programmspeicher und 1 KB RAM-Arbeitsspeicher recht bescheiden aber vollkommen ausreichend. Eine Taktfrequenz von 4 MHz bildet einen guten Kompromiss zwischen Energiesparen und Rechengeschwindigkeit. Zudem ist das Bauteil gut erhältlich und liegt in der Preisklasse der Ein-Euro-μC.

Die Datenübertragung auf Ebene eins erfolgt über einen CAN-Bus. Dieser Feldbus zeich-

[1] http://www.atmel.com

net sich durch seine Robustheit, Priorisierungsmöglichkeiten, sowie einem geringen Strombedarf aus. Die Firma Microchip[2] vertreibt einen preiswerten CAN-Controller unter dem Namen MCP 2515. Mittels SPI-Interface lässt er sich leicht an den ATmega anschließen. Eine weitere energiesparende Maßnahme ist die Möglichkeit, den Mikrocontroller und den Controller während Rechenpausen in einen so genannten Schlafmodus zu versetzen. Die Stromaufnahme kann in dem Fall um den Faktor eintausend gesenkt werden.

Ganz nach dem Prinzip der Benutzerfreundlichkeit erfolgt der Anschluss der Sensoren auf flexible und generische Art und Weise. In den Sensorknoten sind die zahlreichen Ein- und Ausgänge des μC in Form von sechs analogen Eingängen, einem analogen Ausgang, sowie sieben digitalen Ein- und Ausgängen herausgeführt. Der Anschluss von Sensoren und Aktuatoren kann somit direkt oder mit Hilfe eines Aufsteckmoduls für die Anpassung von Spannungspegeln oder das Erlangen einer galvanischen Trennung erfolgen. Die Kommunikation zwischen Ebene null und Ebene eins basiert auf Spannungs-/Stromsignalen und digitalen logischen Zuständen. Wie viele und welche E/A benutzt werden, ist frei wählbar und vielfältig kombinierbar. Die gute Skalierbarkeit der Sensorknoten bleibt sowohl innerhalb eines Knotens (beispielsweise bei Benutzung sämtlicher E/A) als auch bei der Kaskadierung mehrerer Sensorknoten an einem CAN-Bus jederzeit erhalten. Durch einfaches Plug-and-Play können bis zu 30 Sensorknoten an einem CAN-Bus betrieben werden.

Trotz der geringen Rechenleistung und eingeschränkter Hardwareressourcen werden Sicherheitsaspekte bei der Datenübertragung berücksichtigt. Sämtliche Knoten authentisieren sich, und alle Nachrichten erhalten zur Integritätssicherung einen eindeutigen Fingerabdruck. Somit wird es potenziellen Angreifern erschwert, falsche Nachrichten in das System einzuschleusen bzw. durch so genannte Replay-Angriffe abgehörte Szenarien nachzuahmen. Bewusst wurde auf dieser Ebene auf rechenintensive Verschlüsselungsverfahren verzichtet, um schnelle Reaktionszeiten des Systems zu garantieren und die Hardwareressourcen zu minimieren. Da der CAN-Bus nur lokal Einsatz findet, ist Vertraulichkeit der Datenübertragung nicht notwendig.

In sämtlichen Ebenen kommt der “Keyed-Hash Message Authentication Code” (HMAC [6]) zum Einsatz. Zur Berechnung der benötigten Hashfunktion wird SHA-1 [7] benutzt. Eine fortlaufende 16-Bit breite Transaktionsnummer sowie ein vier Byte langer Fingerabdruck schützen das System vor Angreifern. Bedingt durch das einfach gehaltene und schichtentransparente Protokoll besteht die Möglichkeit, dass sich ab einem gewissen Zeitpunkt eine Kombination aus Transaktionsnummer und Fingerabdruck für eine gleiche Nachricht wiederholen könnte. Um diesen potenziellen Schwachpunkt zu beseitigen, erfolgt die Kommunikation in Sitzungen, wobei die Sitzungsschlüssel in regelmäßigen Zeitabständen aktualisiert werden.

Die Zuverlässigkeit des Systems auf Ebene eins basiert auf den ausgefeilten Fehlererkennungsalgorithmen des CAN-Standards. Erzeugt ein CAN-Knoten permanent falsche und ungültige Nachrichten, wird er automatisch vom CAN-Bus ausgeklinkt. Des Weiteren ist CAN eine kollisionsfreie Übertragungstechnologie. Somit gehen keine Nachrichten verloren.

[2] http://www.microchip.com

Masterknoten: Ebene zwei
Der Masterknoten auf Ebene zwei nutzt den leistungsfähigeren Mikrocontroller ATmega644 von Atmel. Mit 64 KB Flash-Speicher und 4 KB Arbeitsspeicher erfüllt er die an ihn gestellten Aufgaben.

Der Hauptaufgabenbereich des Masterknotens besteht darin, die Nachrichten zwischen den Ebenen eins und drei bidirektional weiterzureichen. Auf der einen Seite hängt der Masterknoten am CAN-Bus und an der anderen Seite erfolgt die Datenübertragung über Ethernet. Im Prinzip fungiert er als Vermittler zwischen den Nachrichten vom CAN-Bus und den Ethernet-Paketen. Um den Overhead bei den viel größeren Ethernet-Paketen so gering wie möglich zu halten, werden mehrere CAN-Nachrichten in einen Ethernet-Frame gepackt. Erst wenn das Paket voll ist, ein interner Timeout abgelaufen ist oder eine Nachricht hoher Priorität verschickt werden muss, wird das Paket versandt. Um zeitkritische Nachrichten nicht zu verzögern, ist es möglich, wichtige Sensordaten oder Steuerbefehle mit einen Prioritäts-Bit zu kennzeichnen. Diese Nachrichten werden dann so schnell wie möglich weitergereicht. Die Sicherheit auf dieser Ebene wird ebenfalls, wie schon weiter oben erklärt, mit HMAC und SHA-1 erreicht.

Bei dem CAN-Controller handelt es sich um den schon erwähnten MCP 2515. Als Ethernet-Controller kommt der ENC 28J60 von Microchip zum Einsatz. Mit einem zusätzlichen PoE-Controller kann die Stromversorgung auf dieser Ebene über Power-over-Ethernet erfolgen. Dies minimiert den Verkabelungsaufwand.

Ein weiteres Merkmal ist die Berücksichtigung des Aspekts der Verfügbarkeit im Systemdesign. Bei Ausfall höherer Schichten oder ihrer Anbindung, z.B. bei Ausfall des Ethernet-Netzes, übernimmt ein einfaches Notfallprogramm wichtige Teilaufgaben des Systems und stellt somit den Erhalt einer minimalen Funktionalität sicher. Um das Schichtenmodell nicht zu verletzen, erfolgt die Implementierung dieses Notfallprogrammes als unabhängige Softwarekomponente innerhalb des Masterknotens. Sobald die Ethernet-Verbindung ausfällt, kümmert sich dieses Notfallprogramm um einen minimalen Teilaufgabenbereich der höheren Schicht. Im Rechenzentrum könnte das zum Beispiel bedeuten, dass auch bei Ausfall des Ethernet-Netzes eine existenzielle Reaktion, z.B. das Herunterfahren von Servern bei Überschreitung eines maximalen Temperaturwertes, erhalten bleibt.

Gebäudeknoten, Gateways und Anwendungen: Ebenen drei und vier
Als Gebäudeknoten kann ein ganz konventionelles, stromsparendes eingebettetes System benutzt werden. Auf diesem kleinen Rechner läuft ein kompaktes Betriebssystem mit einem geeigneten Protokollstapel. Der Gebäudeknoten ist für die Verbindung der Ebenen zwei und vier verantwortlich. Eine auf dem System installierte Middleware ermöglicht es, mit Hilfe von geeigneten Steuerbefehlen Daten aus dem Sensornetz zu erlangen und gewünschte Aktionen auszuführen. Beispielsweise kann das Linux-Betriebssystem zum Einsatz kommen. Sehr hohe Zuverlässigkeit kann durch die doppelte Ausführung dieses Gebäudeknotens erreicht werden. Bei Ausfall einer Einheit bleibt das System dank der anderen Einheit voll funktionsfähig.

Noch eine Ebene höher ist der Steuerungsdienst angesiedelt. Er läuft im Leitstand eines Campus. Auf dieser Ebene können Dienste als Schnittstelle zu unterschiedlichsten Proto-

kollen und anderen Anwendungen betrieben werden, z.B. Webservices oder SNMP. Auf dem gleichen oder anderen Rechnern können graphische Benutzeroberflächen und komplexe Steuerungen implementiert werden. Auch eine Anbindung an das Internet kann erfolgen. Als Verschlüsselungsverfahren kann hier das klassische SSL/TLS eingesetzt werden.

Weitere Informationen zu den Möglichkeiten und der Architektur der Ebenen drei und höher finden sich in [1] und der Diplomarbeit [8]. Das nächste Kapitel befasst sich mit der Bewertung des umgesetzten Versuchnetzes und den somit gesammelten Erfahrungen.

4 Erfahrungen mit Prototypen

Zum Testen und Sammeln von Erfahrungen wurde ein Testaufbau der unteren Ebenen des Netzwerkes bestehend aus drei Sensorknoten und einem Masterknoten vorgenommen. Es zeigte sich, dass der Aufbau wie geplant funktioniert und die Anforderungen gut erfllt werden.

Zur Erhöhung der Praxistauglichkeit wurden nach Abschluss der Entwicklung des ersten Prototypen noch folgende Verbesserungen bei der Sensorknoten-Hardware durchgeführt: Mittels eines Umschalters ist es nun möglich, die Stromversorgung der einzelnen Knoten ein- und auszuschalten. Somit können die Knoten während der Installation und Konfiguration bequem nacheinander aktiviert werden, da ein gleichzeitiges Aktivieren mehrerer neuer Knoten nicht erlaubt ist. Um die Fehlersuche zu vereinfachen, wurden jeweils zwei Status-LEDs angebracht. Diese ermöglichen eine schnelle Rückmeldung über den Betriebszustand (Installationsmodus, Konfigurationsmodus oder Betriebsmodus) des jeweiligen Sensorknotens. Zur Spannungsreglung wurden anfangs Regler vom Typ 7805 eingesetzt. Wegen deren hohen Verlustleistung werden in der neuen Hardware-Revision ausschließlich “Low-Drop”-Spannungsregler verwendet. Dadurch konnte die Spannungsversorgung des Netzes um 2 V gesenkt werden. Ein neues doppelseitiges Layout erleichtert die Bestückung der Platinen. Zusätzlich erfüllen nun auch sämtliche Hardware-Komponenten die RoHS-Richtlinie.

Durch die modulare Programmierung konnten die Anforderungen Schritt für Schritt umgesetzt und einzeln getestet werden. Beispielsweise wurde empirisch ermittelt, dass das Authentifizierungsverfahren in den Sensorknoten die Reaktionszeit des Netzes nicht beeinträchtigt. Von besonderem Interesse waren auch die Untersuchungen des nebenläufigen Datenverkehrs. Um die Monopolisierung des Busses durch einen einzelnen Teilnehmer zu vermeiden, wurden Verhaltensregeln entwickelt und evaluiert. Der Masterknoten erhält die höchste Priorität und erhält somit Vorrang für seine Nachrichten. Niedrigere Prioritäten werden von den angeschlossenen Sensorknoten genutzt. Die gewählte Busgeschwindigkeit ermöglicht die Übertragung von durchschnittlich 1000 Nachrichten pro Sekunde. Das Sendepotenzial eines Sensorknotens liegt hingegen bei maximal 30 Nachrichten pro Sekunde. Das heißt, dass auch im Worst-Case zwischen zwei Nachrichten desselben Knotens die anderen 29 Teilnehmer jeweils auch eine Nachricht verschicken können.

Auf der Ethernet-Seite wurde die Belastungsgrenze des Masterknotens erprobt. Der Mas-

terknoten puffert bis zu 150 CAN-Nachrichten. Ist der Puffer voll, werden die ältesten Nachrichten überschrieben. Während den Testphasen wurde der Master absichtlich mit Paketen geflutet. Die Ethernet-Kommunikation war dadurch erwartungsgemäß schwer beeinträchtigt (dies ist systembedingt), jedoch erwiesen sich der Masterknoten und das darunter liegende CAN-Netzwerk als absolut stabil. Nach Einstellen des Denial-of-Service-Angriffes funktionierte das Netzwerk sofort wieder fehlerfrei.

5 Zusammenfassung

Dieser Beitrag beschäftigte sich mit der Verwirklichung eines Sensorknotennetzes zum Messen, Steuern und Schalten. Zu Beginn wurde eine Vielzahl von Anforderungen identifiziert. Dazu gehören Energieverbrauch und Anschaffungskosten genauso wie vielfältige andere Punkte wie Sicherheit, Skalierbarkeit, Zuverlässigkeit und letztendlich auch Benutzerfreundlichkeit in Installation und Betrieb. Diese Anforderungen können nur durch eine Schichtenarchitektur zufriedenstellend umgesetzt werden. Durch das Schichtenmodell konnte die Komplexität und "Intelligenz" möglichst weit weg von unteren Ebenen heraus in Richtung höherer Ebenen verlagert werden.

Es wurde ein Prototyp entwickelt, der sich hervorragend zum Messen und Schalten in Rechenzentren einsetzen lässt. Der Prototyp zeigt, dass die identifizierten Anforderungen auch praktisch in ein funktionierendes Produkt umgesetzt werden können. Das Ergebnis ist ein kostengünstiges, benutzerfreundliches, zuverlässiges, generisches und sicheres Sensorknotennetz, das flexibel an unterschiedlichste Anforderungen anpassbar ist.

Literatur

[1] Dirk Henrici, Patric de Waha, Paul Müller: Bridging the Gap Between Pervasive Devices and Global Networks. International Symposium on Collaborative Technologies and Systems, Workshop on Distributed Collaborative Sensor Networks, 2008

[2] SCADA. URL: de.wikipedia.org/wiki/Supervisory_Control_and_Data_Acquisition (abgerufen 22.01.2009)

[3] LCN. URL: http://www.lcn.de/ (abgerufen 22.01.2009)

[4] BTNode. URL: http://www.btnode.ethz.ch/ (Stand 2007; abgerufen 22.01.2009)

[5] Michel Steichen: Entwicklung kosteneffizienter Sensorknoten mit sicherer Anbindung an offene Netze. Diplomarbeit, TU Kaiserslautern, 2009

[6] H. Krawczyk, M. Bellare, R. Canetti: RFC 2104 - HMAC: Keyed-Hashing for Message Authentication

[7] D. Eastlake, P. Jones: RFC3174 - US Secure Hash Algorithm 1 (SHA1)

[8] Patric de Waha: Sichere und verlässliche Kommunikation zwischen Low-Cost-Devices und PCs. Diplomarbeit, TU Kaiserslautern, 2007

Technologien im Rechenzentrum

Virtualisierungstechnologien in Grid Rechenzentren

Stefan Freitag

stefan.freitag@tu-dortmund.de

Abstract: Kommerzielle und auch akademische Rechenzentren stehen vor der Einführung der Virtualisierungstechnologie oder nutzen diese bereits auf die ein oder andere Weise. Die Hauptmotivation liegt zumeist in der verbesserten Ausnutzung der vorhandenen Kapazitäten durch z.B. Serverkonsolidierung. Diese Arbeit beschreibt den Einfluß von Virtualisierung auf Grid Rechenzentren und weiterhin Integrationsmöglichkeiten auf zwei Ebenen des Grid Middleware Stacks.
Weiterhin wird ein Ansatz zur Virtualisierungsplattform-übergreifenden Erzeugung von Disk Images skizziert. Im Kontext des Übergangs von der Einreichung einfacher Jobs hin zur Einreichung virtueller Maschinen stellt dies eine wichtige Notwendigkeit dar.

1 Einleitung

Die Idee zur Virtualisierung von Hardware stammt aus den Tagen des Mainframe Computing [Stra59] und erlebte in den letzten Jahren einen enormen Zuwachs an Popularität. Insbesondere im Umfeld akademischer und kommerzieller Rechenzentren sind die Vorteile, die sich durch den Einsatz von Virtualisierung ergeben, von besonderem Interesse. Das Hauptziel liegt zumeist in der Steigerung der Gesamtauslastung der vorhandenen Rechen- und Speicherkapazitäten.
Langfristig werden auch Nutzer durch die Kombination von Grid Computing und Virtualisierung profitieren. So sind Nutzer nicht mehr gezwungen ihre Anwendungen an die heterogenen Ausführungsumgebungen auf den Ressourcen anzupassen, sondern liefern ihre Anwendung in einer virtuellen Maschine an die Ressourcenbetreiber aus.

Diese Arbeit richtet ihren Fokus auf den Nutzen von Plattformvirtualisierung für Grid Rechenzentren. Die hier vorgestellten Betrachtungen sind jedoch vollständig auf Cloud Computing übertragbar, welches sich in den letzten Jahren neben Grid Computing etabliert hat. Das Konzept des Cloud Computings besitzt Verbindungen zum Grid Computing und zu anderen Technologien wie dem Cluster Computing oder allgemein, den verteilten Systemen. Jedoch fehlt noch eine allgemein akzeptierte Definition [Fos08]. Insgesamt stehen Grids und Clouds vor ähnlichen Problemstellungen bei der effizienten und automatisierten Ressourcennutzung.

Zunächst ist festzustellen, dass Virtualisierung oftmals als Synonym für Plattformvirtualisierung verstanden wird. Der Begriff der Virtualisierung ist jedoch viel allgemeiner aufzufassen und gliedert sich in Plattformvirtualisierung [Ram04] und Ressourcenvirtualisierung.

Abschnitt 2 stellt die Virtualisierungsformen vor, danach folgt in Abschnitt 3 eine kurze Einführung in Grid Computing. Bisherige und andauernde Arbeiten auf dem Gebiet der Virtualisierung werden in Abschnitt 4 beleuchtet. Abschnitt 5 geht auf Integrationsmöglichkeiten von Virtualisierung in den Grid Middleware Stack ein. Die Arbeit schließt mit einer Zusammenfassung in Abschnitt 6.

2 Virtualisierung

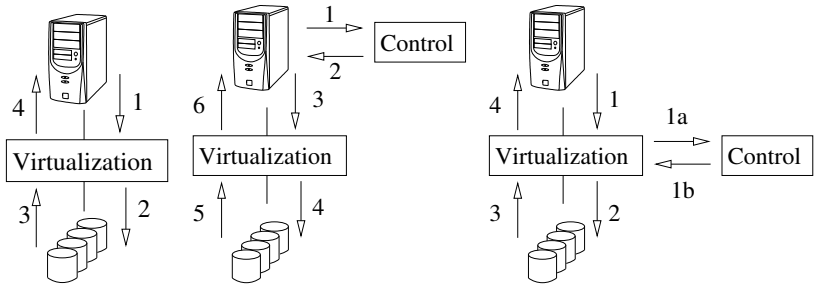
Die Virtualisierungstechnologie fügt eine Abstraktionsschicht in den Hard- und Softwarestack ein. Innerhalb dieser neuen Schicht findet entweder eine Ressourcen-Dekomposition oder -Aggregation statt. Für Nutzer sowie Dienste oberhalb der Virtualisierungsschicht sind dies vollkommen transparente Prozesse. Aktuelle Beispiele für Ressourcenvirtualisierung sind etwa Virtual LANs und Storage Array Networks.

Netzwerkvirtualisierung Auf der Netzwerkschicht spricht man von interner und externer Virtualisierung. Interne Virtualisierung bietet Software-Containern wie virtuellen Maschinen auf einer Ressource eine Netzwerk-ähnliche Funktionalität an. Xen nutzt dieses Konzept um Gästen den Netzwerkzugriff über das Host-System zu ermöglichen [1].
Externe Virtualisierung verschmilzt mehrere Netzwerkpartitionen zu einer einzigen logischen. Bekannte Beispiele für externe Virtualisierung sind VPNs (Virtual Private Networks) [Vpn98] und VLANs (Virtual Local Area Networks) [Vla06]. Jede an einem VPN teilhabende Netzwerkpartition verfügt über einen sog. Gateway. Die Kommunikation zwischen Rechnern in den verschiedenen Netzwerkpartitionen wird über die Gateways getunnelt. Insgesamt erscheinen so die verschiedenen Partitionen für Nutzer innerhalb des VPNs als ein zusammenhängendes, lokales Netzwerk.

Speichervirtualisierung Die Speichervirtualisierung befaßt sich zumeist mit der Aggregation von physikalischem Speicher in größere, logische Einheiten. Hierbei findet Virtualisierung auf unterschiedlichen Ebenen statt: Server Ebene, Fabric Ebene, Storage Subsystem Ebene und Dateisystem Ebene.
Hinsichtlich des Flußes von Kontrollinformation und Daten gliedern sich hier die Virtualisierungsansätze in in-band, out-of-band und split-path [Tat03]. Im Gegensatz zum in-band Ansatz, fließen Kontrollinformation und Daten beim out-of-band (split-path) Ansatz über (teilweise) getrennte Känale (vgl. Abbildung 1). Der wesentliche Unterschied zwischen split-path und out-of-band liegt in der Anordnung der Kontroll-Einheit, welche aus intelligenten SAN-Switches oder speziellen Virtualisierungsappliances besteht.

Plattformvirtualisierung Plattformvirtualisierung reduziert für viele Ressourcenbetreiber langfristig gesehen Kosten wie Wartungs- oder Stromkosten bei gleichzeitiger Steigerung von Redundanz und Dienstgüte. So werden etwa vor anstehenden Wartungsarbeiten

[1] `http://www.cl.cam.ac.uk/Research/SRG/netos/xen/`



(a) In-band (b) Out-of-band (c) Split-path

Abbildung 1: Kontroll- und Datenfluß bei Speichervirtualisierung

an physikalischen Hosts darauf laufende virtuelle Maschinen verschoben. Aus Sicht des Nutzers, dessen Anwendung in einer virtuellen Maschinen ausgeführt wird, ist die Verschiebung ein transparenter Vorgang [Bha08], eine merkliche Nicht-Verfügbarkeit seiner Anwendung entsteht nicht.
Die Nutzung der Plattformvirtualisierung birgt auch Nachteile. Die gesamte Kommunikation der virtuellen Maschine mit der Außenwelt durchläuft die Virtualisierungsschicht, wodurch sich Performanz-Einbußen von bis zu 20% ergeben können [Tat06].

Standards in der Plattformvirtualisierung Die Bündelung von virtueller Maschine, Betriebssystem und Anwendung prägt den Begriff der Virtual Appliance [Vmw07a]. Virtual Appliances kapseln zudem Meta-Daten wie eine hersteller-neutrale Beschreibung und plattformabhängige Informationen zur Installation und Konfiguration. 2008 wurde der Open Virtual Machine Format (OVF) Standard [Ovf08a, Ovf08b] veröffentlicht. Dieser ermöglicht eine Hypervisor-neutrale Beschreibung von Mengen virtueller Maschinen, die als eine logische Einheit aufgefasst und aufgesetzt werden.

3 Grid Computing

Der Fokus beim Grid Computing liegt auf dem sicheren Zugriff auf entfernte Ressourcen (Rechenkraft, Software und Daten) in einer dynamischen, heterogenen Umgebung. Nach Foster et al. [Fos02] lassen sich Grids durch drei Eigenschaften charakterisieren: 1) Bereitstellung nicht-trivialer Dienstgüte, 2) Einsatz von offenen und standardisierten Protokollen und Schnittstellen, 3) Koordination von Ressourcen ohne zentralen Kontrolle.
An der Umsetzung einiger Charakteristika in den aktuellen Grid Middlewares wird noch gearbeitet, derzeit rückt der Einsatz von Standards mehr und mehr in den Vordergrund. So orientieren sich UNICORE [Rie05] und gLite hin zur Nutzung standardisierter Schnittstellen wie OGSA-BES [Bes07]. Langfristiges Ziel ist die Steigerung der Grid-Interoperabilität und somit der Job-Austausch über die Grenzen einer Grid Middleware hinaus.
Das Erreichen dieses Ziel scheint realistisch, da sich viele Grid Middlewares auf ein ge-

meinsames Schichtenmodell abbilden lassen. Die einzelnen Schichten sind in Abbildung 2 dargestellt. Die Infrastrukturebene enthält neben Rechen- und Speicher- ebenso Netzwer-

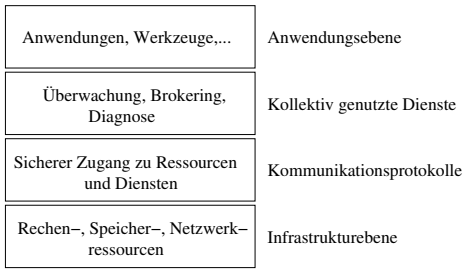


Abbildung 2: Mehr-Schichten Grid-Architektur (in Anlehnung an [Fos03], Seite 47)

kressourcen. Auch Software-Lizenzen können im weiteren Sinne unter dem Ressourcen-Begriff aufgefasst werden. Der Zugriff auf diese lokalen Ressourcen erfolgt über standardisierte Grid-Protokolle wie etwa SRM oder GSI-ssh.
Zudem muß es Grid-Nutzern über Mechanismen erlaubt sein, Aktionen auf der Ressource auszuführen. Im Falle von Rechenressourcen ist eine Job-Kontrolle (Starten/ Stoppen/ Anhalten) sowie -Überwachung sinnvoll. Des Weiteren muß der Transfer von Eingabe- und Ausgabedateien erlaubt sein. Bei Speicherresourcen sind Mechanismen zum Ablegen und Abholen von Dateien notwendig, diese können um Verfahren zur Speicherverwaltung und Advanced Reservation ergänzt werden.

4 Existierende Arbeiten

Plattformvirtualisierung wird im Bereich des Grid Computing vornehmlich zur Server-Konsolidierung und Bereitstellung von HA Diensten verwendet [Car07].
Die Globus Alliance entwickelte den Virtual Workspace Service [Kea05, Fos06], der über eine WSRF-Schnittstelle den Zugriff auf Managementfunktionen virtueller Maschinen erlaubt. Die Funktionalität des Dienstes wurde in [Aga07] gezeigt. Es wird jedoch nur Xen unterstützt.
Die anwendungsabhängige, dynamische Cluster Re-Konfiguration ist in [Eme06] untersucht. Hierbei wurde das LRMS (Local Resource Management System) modifiziert, so dass der Batchsystem Server bei Bedarf Xen-basierte virtuelle Maschinen zu der Liste der verfügbaren Ressourcen hinzufügen bzw. von ihr entfernen kann.

5 Einsatz von Virtualisierung im Grid Computing Umfeld

Die Einbringung der Virtualisierung in Grid Computing ist auf verschiedenen Ebenen möglich. Im Folgenden werden Ansätze auf der Grid Middleware und der LRMS Ebene vorgestellt. Der darauf folgende Abschnitt skizziert eine Lösung zur flexiblen Erstellung

von Virtual Appliances. Dies ist insofern wichtig, da das LRMS bedarfsweise virtuelle Maschinen mit unterschiedlichsten Kombinationen von Betriebssystem und Anwendung bereitstellen muss.
Unabhängig von der untersuchten Ebene ist der Begriff des (Batch-)*Jobs* zu erweitern. Bisher enthalten Jobs auszuführende Shell-Skripte oder sind in einer wohl-definierten Beschreibungssprache wie JDL [Jdl06] oder JSDL [Jsd07] gehalten. Durch die Integration der Plattformvirtualisierung auf LRMS Ebene werden auch virtuelle Maschinen bzw. Virtual Appliances als Jobs ausführbar. Der Unterschied zwischen Job und virtueller Maschine scheint in diesem Kontext zu verschwinden.

5.1 Grid Middleware Ebene

Grid Middleware besitzt, wie jede andere Software auch, Abhängigkeiten, die spätestens während der Installation aufzulösen sind. Diese Abhängigkeiten können sich auf das Betriebssystem oder auf andere Software (Bibliotheken, ...) beziehen. So hat die gLite Middleware starke Abhängigkeiten zu Scientific Linux 4, die UNICORE Middleware ist durch die Ausführung in einer Java VM betriebssystemunabhängig, setzt aber ein installiertes JDK voraus.
Insbesondere im Fall der Betriebssystemabhängigkeit zeigen Virtual Appliances ihren Charme. Die Anwendungen, hier Grid Middleware Dienste, werden mit dem empfohlenen Betriebssystem zu einer Virtual Appliance gebündelt. Das Softwaremanagement wie etwa das Auflösen der Anwendungsanforderungen liegt im Verantwortungsbereich des Virtual Appliance Erzeugers. Ressourcenbetreiber stellen nur noch die Virtualisierungsplattform zur Ausführung der Appliances bereit und übernehmen Managementaufgaben.
Eine Erweiterung des Konzepts stellt die Erzeugung von (Grid Middleware) Diensten durch Nutzer dar. In diesem dynamischen Szenario werden die Managementaufgaben bezüglich der Virtual Appliances vom LRMS übernommen. Dies erhöht den Grad der Automation im Cluster und ermöglicht z. B. den parallelen Betrieb von zwei oder mehr Grid Middlewares auf einer Ressource - ohne gegenseitige Beeinflußung und ohne Eingriff des Ressourcenbetreibers.
Unabhängig davon, ob das Management der Virtual Appliances durch das LRMS oder händisch erfolgt, müssen erforderliche Virtual Appliances vor der Ausführung lokal auf der Ressource vorliegen. Es ist jedoch nicht unbedingt zwingend diese alle lokal vorzuhalten. Storage Elemente, wie sie in den meisten Grid Middlewares existieren, z.B. dCache [2] in der gLite Middleware, bieten sich als Repository an. Der Zugriff auf die Storage Elemente erfolgt über standardisierte Protokolle wie OGSA-DAI in der Globus Toolkit Middleware [3] oder SRM in der gLite Middleware.
Im Bereich des Grid Computing spielt neben der Plattformvirtualisierung ebenso Speichervirtualisierung eine große Rolle. Die Kosten pro Megabyte Speicher fielen in den letzten Jahren kontinuierlich, so dass Ressourcenbetreiber ihre Speicherkapazität drastisch erhöhen konnten. Der Verwaltungsaufwand stieg jedoch überproportional zur hin-

[2] `http://www.dcache.org`
[3] `http://www.globus.org/toolkit/`

zugewonnenen Kapazität an. Die Virtualisierung von Speicher reduziert die Komplexität durch Übernahme von Managementaufgaben. Neben Betreibern profitieren auch Nutzer durch das vereinfachte Datenmanagement. Die Entwicklungen im diesem Bereich sind noch nicht weit fortgeschritten. Langfristig lässt sich erkennen, dass schnellere lokale und Wide-Area-Netzwerke den tatsächlichen Datenstandort zu einer vernachlässigbaren Größe werden lassen.

5.2 LRMS Level

Die Integration der Plattformvirtualisierung inklusive zugehöriger Vorteile wie Checkpointing, Snapshoting und Migration von virtuellen Maschinen in existierende LRMS ist ein aktives Forschungsfeld im Bereich des Grid Computing. Üblicherweise unterstützen LRMS Job Suspension und Checkpointing. Plattformvirtualisierung bietet gleiches im Kontext virtueller Maschinen, tatsächlich bietet sie mit Migration (live oder stop-n-copy) mehr.
Die Kombination der LRMS Eigenschaften mit der Migration virtueller Maschinen erlaubt die dynamische Änderung der Ressourcenallokation. LRMS schauen nicht in die Zukunft, sondern streben für die aktuelle Situation eine optimale Lösung an. Die Auslastung der Ressource ändert sich jedoch dynamisch; einmal getroffene LRMS Entscheidungen können sich zu einem späteren Zeitpunkt als nicht mehr effizient erweisen. Mittels Checkpointing und Suspendierung virtueller Maschinen läßt sich die getroffene, aktuell nicht mehr optimale Ressourcenallokation aufheben.
Die freigewordenen physikalischen Ressourcen werden der Liste der verfügbaren Ressourcen hinzugefügt und die Jobs (virtuelle Maschinen) bis zur weiteren Abarbeitung in einen hold (suspended) Status versetzt. Das LRMS verteilt die Jobs neu auf die verfügbaren Ressourcen und erzielt eine effizientere Lösung. Während der Suspendierung als auch im laufenden Betrieb virtueller Maschinen sind Parameter wie die Anzahl zugewiesener CPUs oder RAM re-konfigurierbar. Somit läßt sich ebenso die Dienstgüte dynamisch anpassen.

5.3 Flexible Erzeugung von Virtual Appliances

Wie in Abschnitt 2 beschrieben handelt es sich bei Virtual Appliances um die Bündelung einer virtuellen Maschine mit einem Betriebssystem und einer spezifischen Anwendung. Einem Rechenzentrum reicht die Existenz einer Virtual Appliance für die Virtualisierungsplattform, die vom Rechenzentrum unterstützt wird, aus. Nutzer, die ihre Virtual Appliances auf mehreren Rechenzentren (mit unterschiedlichen Virtualisierungsplattformen) ausführen wollen, wie es im Grid Computing der Fall ist, stehen vor einem Problem. Sie brauchen ein Werkzeug, welches ein von ihnen erstelltes Festplattenabbild inklusive Betriebssystem und Anwendung in die verschiedenen Formate der Virtualisierungsplattformen transformiert.

Für den Compute Cluster des DGRZR (D-Grid Ressourcen Zentrum Ruhr) ist ein Mischbetrieb von Xen (Workernodes) und VMware (Grid Middleware Dienste) geplant. Der Einsatz von VMware soll die bisher Xen-basierten virtuellen Maschinen für die Grid Middleware Dienste ablösen, um eine Hochverfügbarkeit dieser Dienste zu ermöglichen. Zudem liegt - bedingt durch die Umsetzung der D-Grid Referenz-Installation [4] - mit Scientific Linux 4.x und SUSE Enterprise Linux Server ein Mix an Betriebssystemen vor. Vor diesem Hintergrund wurde Software evaluiert, die Virtual Appliances mit mehreren Betriebssystemen erzeugen kann und mindestens Xen und VMware als Virtualisierungstechnologien unterstützt. Eine Lösung bietet das KIWI Image System [Sch08].

KIWI gestaltet den Erzeugungsprozess eines virtualisierungsplattformabhängigen Abbildes zweistufig. In der ersten Phase wird für das Abbild ein neues Wurzelverzeichnis angelegt. Darin werden aus zuvor definierten Quellen Betriebssystempakete und optional Anwendungspakete eingespielt. Das so gefüllte Wurzelverzeichnis wird als *physical extend* definiert. Am Ende der ersten Phase sind über einen Hook eigene Änderungen wie das De-/Aktivieren von Diensten vornehmbar.

Im zweiten Schritt wird aus dem physical extend ein *logical extend* erzeugt. Dieser logical extend ist plattformspezifisch und besteht z.B. für Xen aus einer Konfigurationsdatei, Disk Image(s) sowie einem initrd Image und einem Kernel. Unabhängig von der als Ausgabeformat gewählten Virtualisierungsplattform wird immer das gleiche Wurzelverzeichnis verwendet. Für Nutzer liegt hierin der wesentliche Vorteil: einmaliges Vorbereiten des Wurzelverzeichnisses und Portabilität auf viele der heute existierenden Virtualisierungsplattformen. Neben den Plattformen Xen und VMware sind auch Live CD/ DVD/ USB Stick als weitere Ausgabeformat möglich. Im Folgenden wird der bisherige Stand der Evaluation kurz beschrieben. KIWI ist zunächst beschränkt auf die Erstellung von Virtual Appliances mit den Betriebssystemen openSUSE, SUSE Linux Enterprise Desktop (SLED) bzw. Server (SLES). Erste Adaptionsschritte zur Unterstützung von Scientific Linux 4, wie es auf dem Compute Cluster zum Einsatz kommt, wurden erarbeitet.

- smart [5], einer der beiden von KIWI unterstützen Paketmanager, wurde auf Scientific Linux 4 sowie 5 übersetzt. smart verarbeitet mehr Paketformate als dessen Alternative zypper [6] und wurde daher präferiert.
- Nach der Erstellung des physical und des logical extends werden über Hooks Funktionen ausgeführt, die SUSE spezifische Anpassungen vornehmen. Für Scientific Linux sind entsprechende Funktionen zu erstellen. Eine Integration neuer Funktionen in KIWI ist aufgrund des verwendeten Namenschemas ohne weiteres möglich: der Name einer Funktion beginnt mit einem Kürzel, das die Linux Distribution beschreibt.
- Für Scientific Linux 4.5, 4.7 und 5.2 konnten bereits physical extends in 32 Bit und 64 Bit Varianten erstellt werden.

[4] `http://dgiref.d-grid.de/wiki/Introduction`
[5] `http://labix.org/smart`
[6] `http://en.opensuse.org/Zypper`

Nach Abschluss der Evaluation soll ein Prozess, zur Integration weiterer Betriebssysteme in KIWI, realisiert sein.

6 Zusammenfassung und Ausblick

Die Arbeit beschreibt Vorteile des Zusammenspiels von Plattformvirtualisierung und Grid Computing aus Sicht von Grid Rechenzentren. Der Trend in den Rechenzentren zur Virtualisierung (sowohl Plattform- als auch Speicher- und Netzwerkvirtualisierung) überzugehen, wurde nicht durch Grid Communities, sondern durch das Bestreben der effizienten Nutzung vorhandener Kapazitäten vorangetrieben. Somit ist Virtualisierung in diesem Kontext keine kurzfristige Erscheinung, ihr Verbreitungsgrad in Rechenzentren wird zunehmen. Virtualisierungstechnologien sind auf verschiedenen Ebenen der Grid Middleware integrierbar, auf zwei von ihnen wurde näher eingegangen. Kommerzielle Anbieter wie Cluster Ressources bieten auf Ebene des lokalen Ressourcemanagements bereits Lösungen für die on-Demand Bereitstellung von Virtual Appliances, jedoch sind diese Lösungen virtualisierungsplattformabhängig.

Ein Vorgehen diese Abhängigkeit zu umgehen wurde ebenso vorgestellt. Das zugehörige Framework KIWI befindet sich derzeit in der Evaluationsphase am DGRZR. Nach Abschluss der Evaluationsphase und der Integration der Unterstützung von Scientific Linux in KIWI, soll über die bereitgestellten Hooks die automatisierte Installation von Grid Middleware Diensten erfolgen. Hierzu wurden Skripte erstellt, die die Installation und auch Konfiguration vieler Dienste der Grid Middlewares gLite3, Globus Toolkit4 und UNICORE5 ermöglichen. Die Kombination der Skripte mit KIWI läßt die Verfügbarkeit von virtualisierungsplattformunabhängigen Grid Middleware Virtual Appliances bedeutend näher rücken.

Literatur

[Aga07] Deploying HEP Applications Using Xen and Globus Virtual Workspaces. A. Agarwal, A.Charbonneau, R. Desmarais, R. Enge, I. Gable, D. Grundy, A. Norton, D. Penfold-Brown, R. Seuster, R.J. Sobie, D.C. Vanderster. In Proceedings of Computing in High Energy and Nuclear Physics. September 2007.

[Bes07] OGSA Basic Execution Service Version 1.0. I. Foster, A. Grimshaw, P. Lane, W. Lee, M. Morgan, S. Newhouse, S. Pickles, D. Pulsipher, C. Smith, M. Theimer. `http://forge.gridforum.org/projects/ogsa-bes-wg`. Letzter Zugriff: 15. Juli 2008.

[Bha08] Virtual Cluster Management with Xen. N. Bhatia, J. S. Vetter. In Lecture Notes in Computer Science, Volume 4854/2008, Seiten 185-194. 2008.

[Car07] Management of a Grid Infrastructure in GLITE with Virtualization. M. Cardenas, J. Perez-Griffo, M. Rubio, R. Ramos. 1st Iberian Grid Infrastructure Conference, Mai 2007.

[Eme06] Dynamic Virtual Clustering with Xen and Moab. W. Emeneker, D, Jackson, J. Butikofer, D. Stanzione. International Symposium on Parallel and Distributed Processing and Applications (ISPA) Workshops 2006. September 2006.

[Fos02] What is the Grid? A Three Point Checklist. I. Foster. 2002

[Fos03] The Grid 2: Blueprint for a New Computing Infrastructure (The Morgan Kaufmann Series in Computer Architecture and Design). I. Foster, C. Kesselman. November 2003.

[Fos06] Virtual Clusters for Grid Communities. I. Foster., T. Freeman, K. Keahey, D. Scheftner, B. Sotomayor, X. Zhang. In Proceedings of the 6th International Symposium on Cluster Computing and Grid (CCGRID). 2006.

[Fos08] Cloud Computing and Grid Computing 360-Degree Compared. I. Foster, I. Yong Zhao Raicu, S. Lu. In Grid Computing Environments Workshop, 2008. Seiten 1 - 10.

[Jdl06] Job Description Language Attributes Specification for the gLite Middleware Version 0.8. F. Pacini. `https://edms.cern.ch/file/590869/1/`. Letzter Zugriff: 15. Juli 2008.

[Jsd07] Job Submission Description Language (JSDL) Specification, Version 1.0. A. Anjomshoaa, M. Drescher, D. Fellows, A. Ly, S. McGough, D. Pulsipher, A. Savva. `http://www.gridforum.org/documents/GFD.56.pdf`. Letzter Zugriff: 15 Juli 2008.

[Kea05] Virtual Workspaces: Achieving Quality of Service and Quality of Life in the Grid. K. Keahey, I. Foster, T. Freeman, X. Zhang. Scientific Programming Journal - Special Issue: Dynamic Grids and Worldwide Computing, Volume 13, No. 4, Seiten 265-276. 2005.

[Ram04] Virtualization - Bringing Flexibility and New Capabilities to Computing Platforms. R. Ramanathan, F. Bruening. Technical Paper. Intel Corporation. 2004

[Ovf08a] The Open Virtual Machine - Whitepaper for OVF Specification Version 0.9. VMware, XenSource. 2007.

[Ovf08b] OVF - Open Virtual Machine Specification Version 0.9. VMware, XenSource. 2007.

[Rie05] Standardization Processes of the UNICORE Grid System. M. Riedel, D. Mallmann. In Proceedings of 1st Austrian Grid Symposium 2005, Seiten 191-203. 2005.

[Sch08] openSUSE - KIWI Image System Cookbook. M. Schäfer. Version 3.01. 24. November 2008

[Stra59] Time sharing in large fast computers. C. Strachey. In Proceedings of the International Conference on Information Processing, UNESCO, Seiten 336-341. 1959.

[Tat03] Virtualization in a SAN. J. Tate. RedBooks Paper, IBM. `http://www.redbooks.ibm.com/redpapers/pdfs/redp3633.pdf`. Letzter Zugriff: 15. Juli 2008.

[Tat06] Making Wide-Area, Multi-Site MPI Feasible Using Xen VM. M. Tatezono, N. Maruyama, S. Matsuoka. International Symposium on Parallel and Distributed Processing and Applications (ISPA) Workshops 2006. September 2006.

[Vla06] IEEE 802.1: 802.1Q - Virtual LANs. IEEE Computer Society. `http://www.ieee802.org/1/pages/\802.1Q.html`. Letzter Zugriff: 15. Juli 2008.

[Vpn98] A Comprehensive Guide to Virtual Private Networks, Volume I: IBM Firewall, Server and Client Solutions. T. Bourne, T. Gaidosch, C. Kunzinger, M. Murhammer, L. Rademacher, A. Weinfurter RedBooks Paper, IBM. `http://www.redbooks.ibm.com/redbooks/pdfs/sg245201.pdf`. Letzter Zugriff: 09. März 2009.

[Vmw07a] Best Practices for Building Virtual Appliances - Whitepaper. VMware. 2007

<myJAM/> – Accounting und Monitoring auf Rechenclustern

Stephan Raub, Dennis-Bendert Schramm, Stephan Olbrich

Lehrstuhl für IT-Management, Institut für Informatik /
Zentrum für Informations- und Medientechnologie (ZIM)
Heinrich-Heine-Universität Düsseldorf
Universitätsstr. 1
40225 Düsseldorf
raub@uni-duesseldorf.de
dennis-bendert.schramm@uni-duesseldorf.de
olbrich@uni-duesseldorf.de

Abstract: Im Rahmen einer Kooperation der Heinrich-Heine-Universität Düsseldorf (HHU) mit der Bull GmbH auf dem Gebiet des High Performance Computing werden Anforderungen an das homogene Management von heterogenen Clusterkonfigurationen analysiert und neue Lösungsansätze entwickelt. Ein Ergebnis des Projekts wird im Papier dargestellt: die prototypische Implementierung eines netzverteilten, modularen, portablen und über eine Weboberfläche interaktiv nutzbaren Systems für das Accounting und Monitoring auf Rechenclustern mit integriertem Projekt- und Anwendungsmanagement. Das Werkzeug *<myJAM/>* („Job Accounting und Monitoring“, `www.myjam.uni-duesseldorf.de`) unterstützt bereits den batchbasierten Betrieb des zentralen heterogenen Linux-Rechenclusters an der HHU unter Linux mit einer Anbindung an das Batchsystem PBS Pro und soll mittelfristig als plattformübergreifende Open-Source-Software bereit gestellt werden.

1 Einleitung

In den letzten Jahren haben sich für das Hochleistungsrechnen Cluster-Lösungen stark verbreitet, die auf kostengünstigen Komponenten der Massenproduktion basieren. Über Interprozess- und Managementnetzwerke wird sowohl die Parallelisierung von Anwendungen als auch ein koordinierter Betrieb unterstützt. Während Cluster in der Vergangenheit meist auf einem homogenen Design vieler gleichartigen Rechenknoten basierten, besteht inzwischen zunehmend der Bedarf, verschiedenartige Rechnerarchitekturen zu integrieren und somit unter einer homogenen Betriebs- und Anwendungsumgebung wahlweise heterogene Systemkomponenten zu nutzen. Darüber hinaus kommen Hybridrechner zum Einsatz, in denen herkömmliche Prozessoren mit Spezialprozessoren, z. B. Compute-Beschleunigern wie Nvidia Tesla GPU-Server, kombiniert werden.

Um die Anwendungen optimal über die Ressourcen des Clusters zu verteilen, werden in der Regel Batchsysteme genutzt. Diese abstrahieren alle technischen Details, wie z. B. das

Finden geeigneter freier Knoten, auf diesen die Anwendungen zu starten, dann zu überwachen, etc. So ist ein Detailwissen über diese komplexen Prozesse für die Nutzer eines Clusters nicht zwingend notwendig, da ein solches Batchsystem eine uniforme Sicht auf den Cluster bietet. Darüber hinaus werden über ein Batchsystem site-spezifische Regeln (die sog. Policies) etabliert, wie z. B. user- oder gruppenspezifische Beschränkungen der kumulativ genutzten Ressourcen. In der Umkehrung können anderen Gruppen exklusive Nutzungsrechte über Ressourcen gewährt werden.

Allerdings werden noch komfortable Werkzeuge benötigt, die mittels Accounting und Monitoring zum effizienten Betrieb, zur Analyse der Nutzung und zur proaktiven Unterstützung der Anwendungsprojekte sowie gegebenenfalls auch zur Abrechnung beitragen.

1.1 Accounting

Unter dem Schlagwort *Accounting* verstehen wir hier die Erfassung sowie die Nutzer-, Job- bzw. Projekt-bezogene Zuordnung der genutzten Ressourcen als Grundlage für eine Berechnung der Kosten der erbrachten IT-Leistungen. Dabei kann es sich auch um rein *virtuelle Kosten* handeln, die alleine für statistische Zwecken erhoben werden, z. B. um die Ressourcennutzung von verschiedenen Institutionen, die einen Cluster gemeinsam nutzen, quantifizieren zu können. Neben der Abrechnungsmöglichkeit stellen auch die Ressourcenplanung und der Einfluss auf die Kontingentierung Motivationen dar.

1.2 Monitoring

In Rahmen des Projekts verstehen wir unter *Monitoring* die unmittelbare systematische Erfassung und Visualisierung der Ressourcennutzung auf einem Cluster. Monitoring ist ein Schlüsselelement, um eine Datenbasis für eine Bewertung und nötigenfalls eine Optimierung der Effizienz eines Systems zu haben. Auch um das Greifen von Policies validieren zu können, ist ein detailliertes und aussagekräftiges Monitoring unerlässlich.

2 Analyse der Anforderungen und Defizite verfügbarer Tools

Zur Unterstützung des Betriebs von Rechenclustern sowie der Anwendungen werden Accounting- und Monitoring-Werkzeuge benötigt, die den folgenden Zwecken und Anforderungen genügen: batchorientierte Ressourcenerfassung, optional prozessorientiertes Accounting, Integration heterogener, netzverteilter Konfigurationen, hoher Interaktionsgrad über eine Weboberfläche, Plattformunabhängigkeit bzw. Portierbarkeit sowie zumindest rudimentäre Verwaltung von Projekten bzw. deren Metadaten.

Die verfügbaren Varianten von Monitoring-Tools sind vielfältig: von kostenlosen Open-

Source-Lösungen bis hin zu hochpreisigen kommerziellen Systemen. Diese Tools überwachen zumeist die einzelnen Knoten eines Clusters und zeigen jeweils Daten über Auslastung, Fehler, etc. an – sie bieten also eine Node-basierte Sicht auf den Cluster.

Die meisten Batchsysteme bringen darüber hinaus auch eigene Tools mit, die ein Monitoring auf Grundlage der dem Batchsystem bekannten Daten anbieten. Die Batchsysteme überwachen die einzelnen Jobs und über die Jobs die von diesen genutzten Knoten eines Clusters – sie bieten also eine Job-basierte Sicht auf den Cluster. Die Tools, die mit einem Batchsystem ausgeliefert werden, bieten z. T. zusätzlich auch eine Node-basierte Sicht. Doch arbeiten diese in der Regel auch nur mit „ihrem“ Batchsystem zusammen. Auch auf die Heterogenisierung gehen die verfügbaren Tools bisher nur geringfügig ein.

3 Grundideen zu einem eigenen Lösungsansatz

Aufgrund der Unzulänglichkeiten der für Accounting und Monitoring verfügbaren Werkzeuge im Vergleich mit den Anforderungen im Rahmen des Betriebs heterogener, kontinuierlich sich verändernder Rechencluster wurde eine Eigenentwicklung begonnen.

Ursprünglich war eine Abspaltung und Weiterentwicklung („Fork“) des Open-Source-Projektes „myPBS“ [MYP] geplant, welches jedoch ab Version 0.8.6 vom 05. April 2006 nicht mehr weiterentwickelt worden ist. Mittlerweile wurde mit dem Werkzeug <myJAM/> jedoch eine komplette Neuentwicklung durchgeführt, so dass es mit seinem „Ur-Vater“ myPBS praktisch nichts mehr gemein hat, außer einigen grundsätzlichen Konzepten, wie z. B. die Verwendung von Perl, MySQL und einem Web-Frontend oder einige Terminologien wie die *Service Unit (SU)* oder die Projekte. In myPBS existierten keine klar voneinander getrennten Programmier-Schichten, so dass durch viele interne Abhängigkeiten der Code schwer wartbar vorgefunden wurde. Darüber hinaus war das Web-Frontend nicht interaktiv und benutzte viele veraltete oder proprietäre Bibliotheken. Moderne Web-Konzepte wie AJAX fehlten, und es wurde keinerlei W3C-Standard eingehalten. All diese Mankos wurden im Zuge der Neuentwicklung ausgemerzt.

<myJAM/> versteht sich als Monitoring- und Accounting-Tool, das mit einem oder mehreren – gegebenenfalls auch verschiedenen – Batchsystemen zusammen arbeitet. Es führt alle bereits verfügbaren und selbst ermittelten bzw. eingegebenen Informationen über Jobs und Knoten, die es vom Batch- oder Betriebssystem bekommt, bzw. Angaben zu den Projekten in Echtzeit zusammen und speichert diese in einer Datenbank. Von dort aus können sowohl Echtzeit- als auch historische Analysen und Darstellungen nach verschiedensten Kriterien über ein hoch interaktives Web-Frontend abgerufen werden. Unschätzbarer Vorteil dieses webbasierten Frontends ist die clientseitige Unabhängigkeit vom verwendeten Betriebssystem. Sowohl homogene als auch beliebig heterogene Cluster werden nativ unterstützt.

Durch das lokale Echtzeit-Monitoring unterscheidet sich <myJAM/> von vielen bereits verfügbaren Lösungen (auch kommerziellen), die lediglich eine Analyse bestehender Log-Files *a posteriori* durchführen. Es ist eine nahe liegende Idee, für das Monitoring Informationen des Batchsystems zu nutzen, wenn ein HPC-Cluster ausschließlich darüber genutzt

wird. Das Batchsystem „weiß“, welche Jobs gerade wo laufen und wie viele Ressourcen von diesen Jobs genutzt werden. Somit liegen schon detaillierte Informationen über die aktuelle Cluster-Auslastung vor und müssen nicht erst aufwändig erneut ermittelt werden.

Das Accounting funktioniert in <myJAM/> projektbasiert. Ein Projekt kann mehrere User enthalten und ein User kann mehreren Projekten angehören. Jedem Projekt kann detailliert der Zugang zu beliebigen Queues des Batchsystems gewährt werden

<myJAM/> macht sich ausschließlich Open-Source-Softwaretechnologien wie PHP [PHP], Perl, Apache [Apa], MySQL [SQL] und OpenFlashChart [Gla] zu Nutze.

Obwohl alle bisherigen <myJAM/>-Installationen unter Linux betrieben werden, sind die genannten Software-Komponenten auch für Windows und MacOS verfügbar. <myJAM/> arbeitet jedoch aktuell nur mit Linux-Batchsystemen (wie z. B. PBS(Pro) [PBS07] oder Torque [TRQ]) zusammen. Auch <myJAM/> wird mittelfristig als Open-Source-Software bereit gestellt.

4 Technische Realisierung: <myJAM/>

<myJAM/> besteht aus drei Hauptkomponenten (siehe Abbildung 1):

- Die <myJAM/>-Datenbank: Eine MySQL-Datenbank, in der alle Informationen über Projekte, User, Jobs und Anwendungen gespeichert werden.
- Der <myJAM/>-Daemon: Ein *nix-Daemon, der auf jedem PBS-Serverknoten läuft. Er sammelt die Informationen über laufende und wartende Jobs vom Batchsystem oder vom Betriebssystem und speichert sie in der Datenbank. Darüber hinaus werden die laufenden Anwendungen klassifiziert.
- Das <myJAM/>-Web-Frontend: Eine hoch-interaktive Web-Applikation, mit dem die gesammelten Informationen in der Datenbank nach verschiedensten Kriterien analysiert und visualisiert werden können. Auch User und Projekte können verwaltet werden.

Jede dieser drei Komponenten (siehe Abschnitte 4.2 bis 4.5) läuft unabhängig von den anderen. Die Kommunikation findet ausschließlich über das standardisierte „MySQL Network Protocol“ statt, weshalb jede Komponente auch auf einem eigenen Server laufen kann. Grundsätzlich können mehrere <myJAM/>-Daemons auf verschiedenen Batch-Servern laufen, um mehrere Batchsysteme überwachen zu können.

Durch die Modularität und die wohldefinierte Kommunikation ist es relativ einfach, neue Daemons für weitere Batchsysteme zu entwickeln. Die Datenbank und das Web-Frontend bedürfen keiner Anpassung. Aus dem gleichen Grund könnten auch einfach neue Frontends entwickelt werden, z. B. ein Kommandozeilen-Interface.

Zusätzlich kommen noch *Prolog*- und *Epilog*-Skripte zum Einsatz. Diese Skripte werden von den meisten Batchsystemen unmittelbar vor (Prolog) und nach (Epilog) dem Start

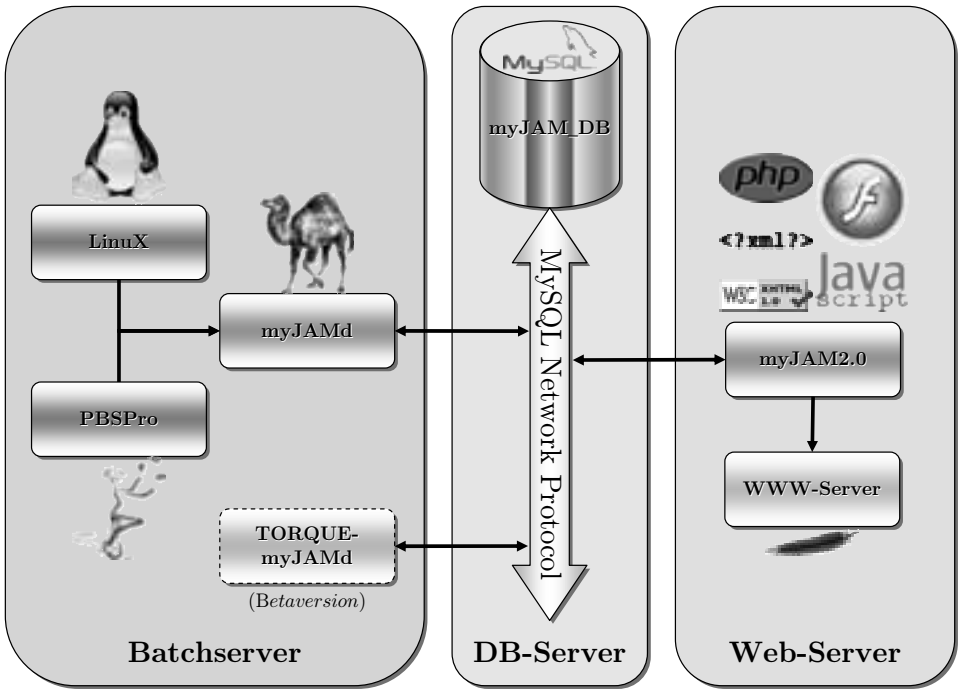


Abbildung 1: Die einzelnen Komponenten von <myJAM/> und deren Interaktionen untereinander

eines Jobs automatisch ausgeführt. <myJAM/> verwendet diesen Mechanismus, um jobbezogene Daten (z. B. Knotennummer, Zeitstempel) in der Datenbank zu speichern.

4.1 Abrechnungsmodelle

<myJAM/> bietet, basierend auf den Accounting-Informationen, die Möglichkeit des *Billings*, also der Berechnung des ökonomischen Wertes der genutzten Ressourcen. Grundeinheit für das Accounting (quasi eine virtuelle Währung innerhalb von <myJAM/>) ist die *ServiceUnit* (*SU*), die einer CPU(Core)-Stunde entspricht. Die Umrechnung der virtuellen Währung *SU* in eine monetäre Währung erfolgt in <myJAM/> über frei definierbare *Kostenmodelle*. Ein Kostenmodell besteht aus zwei Tarifen: dem *Normal-Tarif* und dem *Overrun-Tarif*, jeweils in Währung pro *SU*. Beide Werte können unterschiedlich sein, müssen es aber nicht.

Ein Projekt kann kostenpflichtig oder kostenfrei sein. Ist es kostenpflichtig, können *Credit*- oder *PrePaid*-Kostenmodelle zum Einsatz kommen. Beim Credit-Kostenmodell können User des Projektes *SUs* verbrauchen und nach Beendigung des Projektes oder in regelmäßigen Abständen für die bis dato genutzten *SUs* bezahlen (Normaltarif). Bei PrePaid-

Projekten können User des entsprechenden Projektes nur diejenigen *SUs* verbrauchen, die als *SU*-Guthaben auf dem Konto des Projektes vorhanden sind (durch vorhergehende Einzahlung zum Normaltarif). Darüber hinaus ist es möglich, eine gewisse Überziehung zuzulassen und diese mit der nächsten Einzahlung zu verrechnen (zum Overrun-Tarif), oder eine gesonderte Rechnung dafür auszustellen.

4.2 Die <myJAM/>-Datenbank

Die Datenbank von <myJAM/> wurde mit MySQL5 [SQL] realisiert. Unser Datenbankdesign berücksichtigt weitgehend die *dritte Normalform*, um Dateninkonsistenzen vorzubeugen. Aus Performance-Gründen wurden lediglich einige wenige Tabellen auf die zweite Normalform denormalisiert. User, Projekte, Warteschlangen, Hosts, Anwendungen und Kostenmodelle bilden die zentralen Tabellen. Alle Einträge besitzen Surrogatschlüssel als Primärschlüssel. Attribute, die auf Einträge einer anderen Tabelle referenzieren, benutzen diese Surrogatschlüssel als Fremdschlüssel. Für das Web-Frontend werden die Einträge in den Tabellen und die Referenzen auf andere Tabellen in PHP5-Objekte im Rahmen der <myJAM/>-Klassenbibliothek umgesetzt (*object-relational mapping*).

Mehrere Prozesse greifen auf die Datenbank zu: das <myJAM/>-Web-Frontend, ein oder mehrere <myJAM/>-Daemons und diverse Prolog- und Epilog-Skripte. Daher ist ein rigoroses Locking unbedingt notwendig. Der Nachteil dabei ist, dass jedes Mal, wenn eine oder mehrere Tabellen gelockt sind, alle anderen Prozesse bis zur Freigabe der entsprechenden Tabellen blockiert sind. Zurzeit priorisieren wir absolute Datenkonsistenz, weshalb wir recht ausgiebig vom Locking Gebrauch machen, wohl wissend, welche anderen Probleme (z. B. die Wartezeiten im Web-Frontend) wir damit produzieren. Die Locking-Strategie ist einer der Punkte unserer ständigen Weiterentwicklung.

4.3 Der <myJAM/>-Daemon

Um die Informationen über laufende und wartende Jobs zu sammeln, läuft auf jedem Batchsystem-Server ein <myJAM/>-Daemon-Prozess.

Das Core-Modul des <myJAM/>-Daemon besteht aus einer Schleife über alle (laufenden, wartenden oder angehaltenen) Jobs des Batchsystems. Für laufende Jobs werden vom Batchsystem der aktuell genutzte Arbeitsspeicher, die Walltime und die CPU-Auslastung abgefragt. Zusätzlich ermittelt der Daemon die laufende Anwendung durch Interaktion mit dem Betriebssystem (siehe nächster Abschnitt). Alle gesammelten Informationen werden vom Daemon in die Datenbank geschrieben.

In seltenen Fällen stirbt ein Job, ohne dass das Batchsystem das Epilogskript ausführt, so dass der Job noch in der <myJAM/>-Datenbank als laufender Job geführt wird. Um zu vermeiden, dass sich solche Zombie-Jobs ansammeln, wird in regelmäßigen Abständen eine Garbage-Collection innerhalb des Daemons ausgeführt.

4.4 Anwendungsklassifikation (Software-Accounting) im <myJAM/>-Daemon

<myJAM/> besitzt als ein herausragendes Feature die Fähigkeit, die gerade auf dem Cluster laufenden Anwendungen zu erkennen (*Software-Accounting*). Als Anwendung verstehen wir in diesem Zusammenhang eine ausführbare Datei (Binary), ein Skript oder eine zusammengehörende Sammlung von Binaries und/oder Skripten, die zusammen ein in sich geschlossenes Programm-Paket ergeben.

Die Information, was genau gerade auf welchem Knoten läuft, gehört zu einem umfassenden Monitoring dazu. Die historische Analyse dieser Informationen erlaubt es, Anschaffungen an den Nutzerkreis anzupassen oder gezielt Schulungen für die „Hauptnutzergemeinde“ gezielt anzubieten.

Doch genau hier tun sich die meisten etablierten Tools sehr schwer: Batchsysteme geben hierbei meist nur einen Surrogatschlüssel an (z.B. eine fortlaufende Job-Nummer). Über diesen Surrogatschlüssel sind Zugriffe auf Detailinformationen dieses Jobs möglich, worunter sich auch ein vom User frei wählbarer Jobname befinden kann. Um jedoch zuverlässig anzeigen zu können, welche Anwendung zu diesem Job gehört, wäre man auf die Unterstützung der User angewiesen, die für jeden Job angeben müssen, um was für eine Anwendung es sich handelt. Das ist für die User lästig und insgesamt fehleranfällig. Eine automatische, erweiterbare Erkennung der Anwendung gab es unseres Wissens bisher nicht. <myJAM/> soll diesen Mangel beheben.

Das Betriebssystem kennt den Dateinamen des gerade ausgeführten Binarys oder Skripts. Doch das alleine reicht nicht, da viele Nutzer dazu neigen, jedes ihrer selbst entwickelten Binaries „a.out“ zu nennen, obwohl es sicht *de facto* um völlig unterschiedliche Anwendungen handelt. Daher bietet sich zur eindeutigen Erkennung für Binaries und Skripte eine Hash-Funktion, wie z. B. der Message-Digest-Algorithmus 5 (MD5) von Rivest [Riv92] an. Mit dem MD5-Wert als Schlüssel kann dann aus einer Datenbank abgefragt werden, um welche Anwendung es sich tatsächlich handelt.

Die Hashes der gängigen Binaries und Skripte werden vom Administrator des Clusters in die Datenbank eingepflegt. Unbekannte Hashes (und damit bisher noch nicht erfasste Binaries oder Skripte) werden vorläufig unter ihrem vollen Pfad abgelegt. Der Administrator kann nachträglich die Schlüssel zu bereits bestehenden oder einer neuen Anwendungen hinzufügen. Das Konzept lebt und stirbt damit, wie einfach und komfortabel diese Datenbank aktuell gehalten werden kann. Deshalb haben wir versucht ein durchgängiges Konzept umzusetzen, bei dem man an jeder Stelle des Frontends, an der man mit einem Binary (oder Skript) in Kontakt kommt, durch einen einfachen Klick, den entsprechenden Hash einer Anwendung zuschlagen kann.

4.5 Das <myJAM/>-Web-Frontend

Das <myJAM/>-Web-Frontend ist eine hoch-interaktive Web-Applikation, die für User und Administratoren gleichermaßen das zentrale Interface für das Monitoring von Jobs, Warteschlangen oder des ganzen Clusters, für die Verwaltung von Usern und Projekten, so-

Abbildung 2: Zuordnung von Binaries zu Applikationen für das Software-Accounting in <myJAM/>

wie zur Visualisierung aktueller und historischer Analysen und Statistiken. Das Frontend nutzt neben W3C-konformem XHTML [Pem02] auch objektorientiertes PHP5 [PHP], JavaScript / AJAX und OpenFlashChart [Gla].

4.5.1 Schichtenmodell

Zur Reduzierung von Abhängigkeiten und um die Komplexität möglichst gering zu halten, wurde beim Softwaredesign des <myJAM/>-Web-Frontends ein strenges Schichtenmodell zugrunde gelegt (Abbildung 3).

Backend ist der MySQL-Server. Für den Zugriff auf diesen Server per TCP/IP zeichnet eine eigene Klasse verantwortlich. Sie nimmt MySQL-Anfragen von den Klassen der Schicht darunter entgegen und liefert die Ergebnisse fertig aufbereitet als assoziatives Array zurück. Um Script-Injection-Angriffe abzuwehren, werden alle Sonderzeichen in ihre entsprechenden HTML-Tags konvertiert. Außerdem beinhaltet die Klasse Methoden zur Überprüfung von Usereingaben auf SQL-Injections.

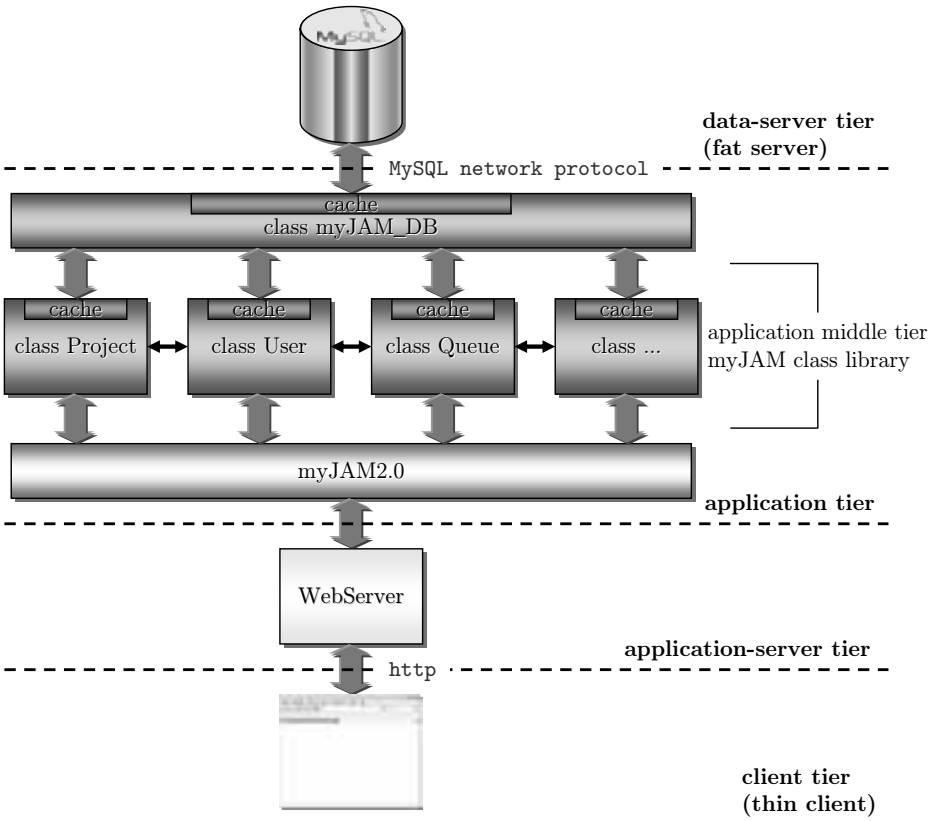


Abbildung 3: Schichtenmodell von <myJAM/>.

4.5.2 Object-Relational Mapping

Die Daten aus den MySQL-Tabellen werden durch *object-relational mapping* (ORM, Objekt-Relationale Abbildung) als PHP5-Objekte von der <myJAM/>-Klassenbibliothek abgebildet. Jedes Objekt eines Typs entspricht dabei einer Zeile derjenigen Tabelle, die diesen Typ repräsentiert. Fremdschlüssel-Primärschlüssel-Beziehungen werden durch Referenzen auf passende Objekte realisiert.

Für die Attribute der Objekte existieren passende *Setters* und *Getters*, in denen direkt erste Sicherheits- und Plausibilitätsüberprüfungen stattfinden. Es können auch komplexe Meta-Attribute, die aus mehreren Einträgen unterschiedlicher MySQL-Tabellen abgeleitet werden, abgefragt werden, z. B. die Anzahl gerade laufender oder wartender Jobs.

4.5.3 Design der Web-Applikation

Ein intuitiv bedienbares und konsistentes Design entscheidet maßgeblich über die Benutzbarkeit einer Anwendung. Das gilt umso mehr bei Web-Applikationen, von denen seitens der Anwender mittlerweile erwartet wird, dass sie selbsterklärend und robust gegen Trial-And-Error-Bedienung sind.

Ein wesentliches Designmerkmal des <myJAM/>-Frontends ist die gegenseitige Verlinkung von Bereichen untereinander (Abbildung 4). Dieses Konzept ermöglicht es, Bereiche und Funktionen auf mehreren Wegen innerhalb von <myJAM/> zu erreichen. User, Projekte, Anwendungen, Jobs, etc. können in jedem Kontext angeklickt werden und führen direkt zu der entsprechenden Detailansicht auf Basis der Methoden dieses Objekts.

Viele Listen- oder Diagrammansichten hängen von Parametern oder Schlüsselwörtern ab. An vielen Stellen wurden AJAX-Methoden implementiert, um diese Ansichten aktualisieren zu können, wenn es eine Useraktion erforderlich macht. So wird die Liste der gelaufenen Anwendungen bereits während der Eingabe des Suchbegriffes aktualisiert.

Die grafische Darstellung von Daten aus der <myJAM/>-Datenbank erfolgt mit Hilfe der Open-Source-Bibliothek „OpenFlashChart“ [Gla]. Diese Bibliothek stellt für verschiedene Diagramm-Typen (wie Linen-, Balken- oder Tortendiagramme) PHP-Objekte zur Verfügung und stellt die Diagramme dann als Flash-Content dar.

4.5.4 Sicherheit

Einer der wichtigsten Grundsätze für Web-Applikationen lautet: Traue nie einer Benutzereingabe. Aus diesem Grunde werden alle Benutzereingabe in <myJAM/> bereits auf Clientseite per JavaScript geprüft – insbesondere auf Script- oder SQL-Injections. Passwörter werden auch bereits auf Clientseite verschlüsselt und dann nur noch das verschlüsselte Ergebnis per POST zum Server geschickt. Trotz der in [WFLY04, WY05] aufgezeigten Kollision der MD5-Funktion erachten wir dieses Verfahren als sicher.

5 Zusammenfassung

Mit <myJAM/> wurde ein leicht und intuitiv bedienbares Frontend für Accounting und Monitoring auf der Basis bereits am Markt etablierter Batchsysteme entwickelt. <myJAM/> wurde in einer heterogenen Cluster-Umgebung entwickelt und erprobt. Durch ein klares Systemdesign und wohldefinierte Schnittstellen können zusätzliche und künftige Batchsysteme sehr leicht unterstützt werden.

<myJAM/> zeigt den aktuellen Status des Clusters in Form von Ressourcen-Auslastung, genutzten Anwendungen – hierfür wurde ein innovatives Software-Accounting implementiert – und in einer Übersicht der Auslastung der verschiedenen Warteschlangen der Batchsysteme. Auch Analysen beliebiger vergangener Zeiträume können durchgeführt werden – jeweils als Verlauf pro Monat oder kumulativ. Durch eigene Kostenmodelle können die

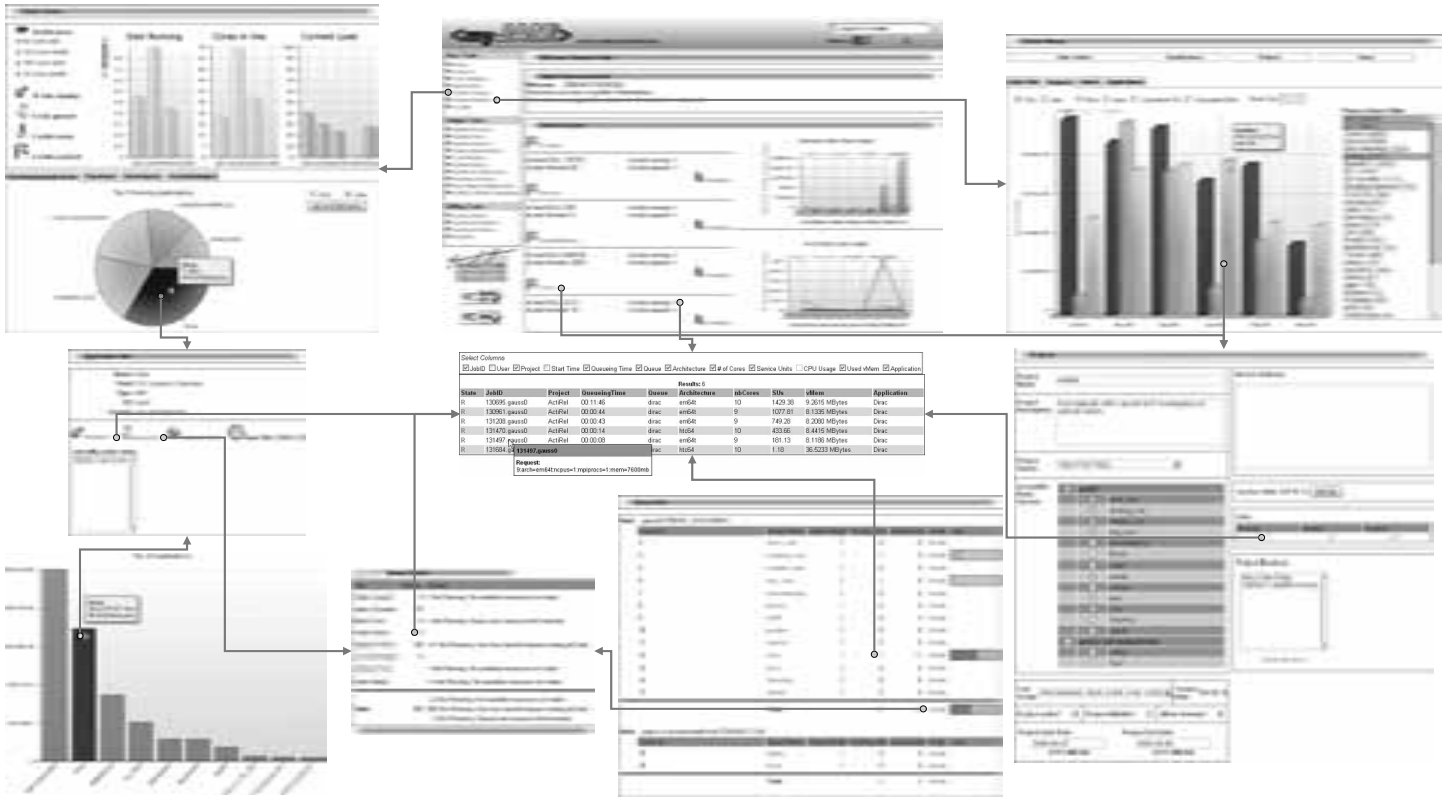


Abbildung 4: Die <myJAM/>-Weboberfläche

von den Projekten genutzten CPU-Stunden monetär abgerechnet oder in Beziehung zu einer Kontingentierung gesetzt werden.

<myJAM/> bietet wertvolle Accounting- und Monitoring-Fähigkeiten zur Unterstützung der Systemadministration, des Nutzersupports und des Berichtswesens. Geplante Erweiterungen sind eine Node-basierte Sicht sowie ein Hochverfügbarkeitsmodul. Darüber hinaus sind weitere Test- und Referenzinstallationen auf großen Clustern geplant, insbesondere in Kooperation mit dem Jülich Supercomputing Centre (JSC).

Danksagung

Die Autoren danken der Firma Bull GmbH für die Projektfinanzierung und die konstruktive Zusammenarbeit sowie dem HPC-Team am ZIM für die Nutzung und betriebliche Unterstützung des Bull-HPC-Clusters zur Entwicklung und Erprobung von <myJAM/>.

References

[Apa] *Apache – HTTP Server Project*. http://httpd.apache.org.

[Gla] John Glazebrook. *The Open Flash Chart project*. http://teethgrinder.co.uk/open-flash-chart.

[MYP] *MyPBS*. http://my-pbs.sourceforge.net.

[PBS07] Altair Engineering, Inc. *PBS Professional 9.0 Administrator's Guide*, 2007.

[Pem02] Steven Pemberton. XHTML™ 1.0 The Extensible HyperText Markup Language (Second Edition). W3C recommendation, W3C, August 2002. http://www.w3.org/TR/2002/REC-xhtml1-20020801.

[PHP] *PHP – A Hypertext Preprocessor*. http://www.php.net.

[Riv92] R. Rivest. The MD5 Message-Digest Algorithm. RFC 1321, Internet Engineering Task Force, April 1992.

[SQL] *MySQL – The world's most popular open source database*. http://www.mysql.com.

[TRQ] *TORQUE Ressource Manager*. http://www.clusterresources.com/products/torque.

[WFLY04] Xiaoyun Wang, Dengguo Feng, Xuejia Lai, and Hongbo Yu. Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD. Cryptology ePrint Archive, Report 2004/199, 2004. http://eprint.iacr.org/.

[WY05] Xiaoyun Wang and Hongbo Yu. How to Break MD5 and Other Hash Functions. In *EUROCRYPT*, pages 19–35, 2005.

GI-Edition Lecture Notes in Informatics

P-1 Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001.

P-2 Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001.

P-3 Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Language to Information Systems, NLDB'2001.

P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensorgestützte Chirurgie; Workshop des SFB 414.

P-5 Andy Schürr (Hg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems.

P-6 Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001.

P-7 Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001.

P-8 Reinhard Keil-Slawik, Johannes Magenheim (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.

P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeitstagung.

P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience.

P-11 Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002.

P-12 Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002.

P-13 Jan von Knop, Peter Schirmbacher and Viljan Mahni_ (Hrsg.): The Changing Universities – The Role of Technology.

P-14 Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Semantic Web – XSW 2002.

P-15 Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine.

P-16 J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Busi-ness 2002.

P-17 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze –Die Verletzbarkeit meistern, 16. DFN Arbeitstagung.

P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2002.

P-19 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund.

P-20 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3.Okt. 2002 in Dortmund (Ergänzungsband).

P-21 Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen.

P-22 Sigrid Schubert, Johannes Magenheim, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation.

P-23 Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit

P-24 Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen

P-25 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003

P-26 Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web

P-27 Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin

P-28 Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Erfahrungen und Visionen

P-29 Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems

P-30 Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications

P-31 Arslan Brömme, Christoph Busch (Eds.): BIOSIG 2003: Biometric and Electronic Signatures

P-32 Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003

P-33 Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft

P-34 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1)

P-35 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2)

P-36 Rüdiger Grimm, Hubert B. Keller, Kai Rannenberg (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik

P-37 Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik

P-38 E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003

P-39 Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003

P-40 Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004

P-41 Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing

P-42 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste

P-43 Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistance, Scalability, Transactions – Database Mechanisms for Mobile Applications

P-44 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning. E-Services

P-45 Bernhard Rumpe, Wofgang Hesse (Hrsg.): Modellierung 2004

P-46 Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment

P-47 Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society

P-48 Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications

P-49 G. Schiefer, P. Wagner, M. Morgenstern, U. Rickert (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven

P-50 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-51 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm

P-52 Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik

P-53 Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004

P-54 Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration

P-55 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science und Grid Adhoc-Netze Medienintegration

P-56 Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government

P-57 Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen

P-58 Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schweiggert (Hrsg.): Testing of Component-Based Systems and Software Quality

P-59 J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Ranneberg, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments

P-60 Steffen Friedrich (Hrsg.): Unterrichtskonzepte für inforrmatische Bildung

P-61 Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen

P-62 Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit

P-63 Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and ist Applications

P-64 Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005

P-65 Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolffried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web

P-66 Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik

P-67 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1)

P-68 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2)

P-69 Robert Hirschfeld, Ryszard Kowalcyk, Andreas Polze, Matthias Weske (Hrsg.): NODe 2005, GSEM 2005

P-70 Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005)

P-71 Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005

P-72 Klaus P. Jantke, Klaus-Peter Fähnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment

P-73 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): “Heute schon das Morgen sehen“

P-74 Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology

P-75 Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture

P-76 Thomas Kirste, Birgitta König-Riess, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz

P-77 Jana Dittmann (Hrsg.): SICHERHEIT 2006

P-78 K.-O. Wenkel, P. Wagner, M. Morgenstern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel

P-79 Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006

P-80 Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce

P-81 Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS´06

P-82 Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006

P-83 Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics

P-84 Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications

P-85 Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems

P-86 Robert Krimmer (Ed.): Electronic Voting 2006

P-87 Max Mühlhäuser, Guido Rößling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik

P-88 Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODe 2006, GSEM 2006

P-90 Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur

P-91 Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006

P-92 Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006

P-93 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1

P-94 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2

P-95 Matthias Weske, Markus Nüttgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen

P-96 Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies

P-97 Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Mangament & IT-Forensics – IMF 2006

P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices

P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik

P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik

P-101 Stefan Böttinger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten

P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments

P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)

P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme

P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007

P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops

P-107 Heinrich C. Mayr, Dimitris Karagiannis (eds.) Information Systems Technology and its Applications

P-108 Arslan Brömme, Christoph Busch, Detlef Hühnlein (eds.) BIOSIG 2007: Biometrics and Electronic Signatures

P-109 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 1

P-110 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 2

P-111 Christian Eibl, Johannes Magenheim, Sigrid Schubert, Martin Wessner (Hrsg.) DeLFI 2007: 5. e-Learning Fachtagung Informatik

P-112 Sigrid Schubert (Hrsg.) Didaktik der Informatik in Theorie und Praxis

P-113 Sören Auer, Christian Bizer, Claudia Müller, Anna V. Zhdanova (Eds.) The Social Semantic Web 2007 Proceedings of the 1st Conference on Social Semantic Web (CSSW)

P-114 Sandra Frings, Oliver Göbel, Detlef Günther, Hardo G. Hase, Jens Nedon, Dirk Schadt, Arslan Brömme (Eds.) IMF2007 IT-incident management & IT-forensics Proceedings of the 3rd International Conference on IT-Incident Management & IT-Forensics

P-115 Claudia Falter, Alexander Schliep, Joachim Selbig, Martin Vingron and Dirk Walther (Eds.) German conference on bioinformatics GCB 2007

P-116 Witold Abramowicz, Leszek Maciszek (Eds.) Business Process and Services Computing 1st International Working Conference on Business Process and Services Computing BPSC 2007

P-117 Ryszard Kowalczyk (Ed.) Grid service engineering and manegement The 4th International Conference on Grid Service Engineering and Management GSEM 2007

P-118 Andreas Hein, Wilfried Thoben, Hans-Jürgen Appelrath, Peter Jensch (Eds.) European Conference on ehealth 2007

P-119 Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.) Enterprise Modelling and Information Systems Architectures Concepts and Applications

P-120 Adam Pawlak, Kurt Sandkuhl, Wojciech Cholewa, Leandro Soares Indrusiak (Eds.) Coordination of Collaborative Engineering - State of the Art and Future Challenges

P-121 Korbinian Herrmann, Bernd Bruegge (Hrsg.) Software Engineering 2008 Fachtagung des GI-Fachbereichs Softwaretechnik

P-122 Walid Maalej, Bernd Bruegge (Hrsg.) Software Engineering 2008 - Workshopband Fachtagung des GI-Fachbereichs Softwaretechnik

P-123 Michael H. Breitner, Martin Breunig, Elgar Fleisch, Ley Pousttchi, Klaus Turowski (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Technologien, Prozesse, Marktfähigkeit
Proceedings zur 3. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2008)

P-124 Wolfgang E. Nagel, Rolf Hoffmann, Andreas Koch (Eds.)
9th Workshop on Parallel Systems and Algorithms (PASA)
Workshop of the GI/ITG Speciel Interest Groups PARS and PARVA

P-125 Rolf A.E. Müller, Hans-H. Sundermeier, Ludwig Theuvsen, Stephanie Schütze, Marlies Morgenstern (Hrsg.)
Unternehmens-IT: Führungsinstrument oder Verwaltungsbürde
Referate der 28. GIL Jahrestagung

P-126 Rainer Gimnich, Uwe Kaiser, Jochen Quante, Andreas Winter (Hrsg.)
10th Workshop Software Reengineering (WSR 2008)

P-127 Thomas Kühne, Wolfgang Reisig, Friedrich Steimann (Hrsg.)
Modellierung 2008

P-128 Ammar Alkassar, Jörg Siekmann (Hrsg.)
Sicherheit 2008
Sicherheit, Schutz und Zuverlässigkeit
Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)
2.-4. April 2008
Saarbrücken, Germany

P-129 Wolfgang Hesse, Andreas Oberweis (Eds.)
Sigsand-Europe 2008
Proceedings of the Third AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems

P-130 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
1. DFN-Forum Kommunikations-technologien Beiträge der Fachtagung

P-131 Robert Krimmer, Rüdiger Grimm (Eds.)
3rd International Conference on Electronic Voting 2008
Co-organized by Council of Europe, Gesellschaft für Informatik and E-Voting.CC

P-132 Silke Seehusen, Ulrike Lucke, Stefan Fischer (Hrsg.)
DeLFI 2008:
Die 6. e-Learning Fachtagung Informatik

P-133 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 1

P-134 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)
INFORMATIK 2008
Beherrschbare Systeme – dank Informatik
Band 2

P-135 Torsten Brinda, Michael Fothe, Peter Hubwieser, Kirsten Schlüter (Hrsg.)
Didaktik der Informatik – Aktuelle Forschungsergebnisse

P-136 Andreas Beyer, Michael Schroeder (Eds.)
German Conference on Bioinformatics GCB 2008

P-137 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)
BIOSIG 2008: Biometrics and Electronic Signatures

P-138 Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau, Klaus Turowski (Hrsg.)
Synergien durch Integration und Informationslogistik
Proceedings zur DW2008

P-139 Georg Herzwurm, Martin Mikusz (Hrsg.)
Industrialisierung des Software-Managements
Fachtagung des GI-Fachausschusses Management der Anwendungsentwick-lung und -wartung im Fachbereich Wirtschaftsinformatik

P-140 Oliver Göbel, Sandra Frings, Detlef Günther, Jens Nedon, Dirk Schadt (Eds.)
IMF 2008 - IT Incident Management & IT Forensics

P-141 Peter Loos, Markus Nüttgens, Klaus Turowski, Dirk Werth (Hrsg.)
Modellierung betrieblicher Informations-systeme (MobIS 2008)
Modellierung zwischen SOA und Compliance Management

P-142 R. Bill, P. Korduan, L. Theuvsen, M. Morgenstern (Hrsg.)
Anforderungen an die Agrarinformatik durch Globalisierung und Klimaveränderung

P-143 Peter Liggesmeyer, Gregor Engels, Jürgen Münch, Jörg Dörr, Norman Riegel (Hrsg.)
Software Engineering 2009
Fachtagung des GI-Fachbereichs Softwaretechnik

P-144 Johann-Christoph Freytag, Thomas Ruf, Wolfgang Lehner, Gottfried Vossen (Hrsg.)
Datenbanksysteme in Business, Technologie und Web (BTW)

P-145 Knut Hinkelmann, Holger Wache (Eds.)
WM2009: 5th Conference on Professional Knowledge Management

P-146 Markus Bick, Martin Breunig, Hagen Höpfner (Hrsg.)
Mobile und Ubiquitäre Informationssysteme – Entwicklung, Implementierung und Anwendung
4. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2009)

P-147 Witold Abramowicz, Leszek Maciaszek, Ryszard Kowalczyk, Andreas Speck (Eds.)
Business Process, Services Computing and Intelligent Service Management
BPSC 2009 · ISM 2009 · YRW-MBP 2009

P-149 Paul Müller, Bernhard Neumair, Gabi Dreo Rodosek (Hrsg.)
2. DFN-Forum Kommunikationstechnologien
Beiträge der Fachtagung

The titles can be purchased at:

Köllen Druck + Verlag GmbH
Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn
Fax: +49 (0)228/9898222
E-Mail: druckverlag@koellen.de