

GWGD-Bericht Nr. 71

Klaus Hermann, Jörg Kantel
(Hrsg.)

**23. DV-Treffen der
Max-Planck-Institute**

**15. - 17. November 2006
in Berlin**

Klaus Hermann, Jörg Kantel (Hrsg.)

23. DV-Treffen der
Max-Planck-Institute

15. - 17. November 2006
in Berlin

Klaus Hermann, Jörg Kantel (Hrsg.)

23. DV-Treffen der Max-Planck-Institute

**15. - 17 November 2006
in Berlin**

GWDG-Bericht Nr. 71

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen

© 2007

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen

Am Faßberg

D-37077 Göttingen

Telefon: 0551 201-1510

Telefax: 0551 201-2150

E-Mail: gwdg@gwdg.de

Satz: Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen

Druck: Artificium Stietenroth, Bodenfelde

ISSN 0176-2516

Inhalt

Vorwort	1
Web 2.0: Werkzeuge für die Wissenschaft <i>Jörg Kantel</i>	3
Langzeitarchivierung digitaler Datenbestände an der Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen (GWDG) am Beispiel des Kunsthistorischen Instituts in Florenz – Max-Planck-Institut (KHI) <i>Timo Scheller</i>	41
Digitale Keilschriftbibliothek lexikalischer Listen aus Assur <i>Frauke Weiershäuser, Thomas Linnemann</i>	51
Web 2.0-Techniken für die CDLI <i>Sebastian Schröder</i>	65

Zukünftige HPC-Systeme der GWDG <i>Oswald Haan</i>	75
Ein System zur Bearbeitung weltweiter Bewerbungen auf Forschungsstellen der Max-Planck-Gesellschaft <i>Wilfried Grieger</i>	91
Zielgenaue Verteilung, Konfiguration und Verwaltung von Software in heterogenen IT-Umgebungen <i>Bertram Smolny, Mario Gzuk, Tino Mai, Hartmut Arlt</i>	103

Vorwort

Der vorliegende Band enthält mehrere Beiträge des 23. DV-Treffens der Max-Planck-Institute, das vom 15. bis 17. November 2006 im Harnack-Haus und in den Räumen des Fritz-Haber-Instituts der MPG in Berlin stattfand.

Traditionsgemäß begann das Treffen mit mehreren parallelen Workshops zu ausgesuchten Themen aus dem Tätigkeitsbereich der DV-Abteilungen, die über Security, Storage-Lösungen bis hin zu Web 2.0 reichten. Danach schlossen sich zwei Tage mit Vorträgen an. Bemerkenswert war in diesem Jahr, dass mehr Vorträge als bei früheren Treffen von Mitarbeiterinnen und Mitarbeitern der DV-Abteilungen aus geisteswissenschaftlichen Instituten gehalten wurden, was sich auch in diesem Band niederschlägt. Weiterhin stellte sich der Leiter der neu gegründeten Gruppe „Max-Planck-Digital-Library“ den Teilnehmerinnen und Teilnehmern des Treffens vor. Außerdem wurde eine Vortrags- und Workshop-Reihe „e-Science Seminare für die MPG“ angeregt, die im Jahr 2007 schon mehrmals stattgefunden hat.

Das jährliche Treffen wird von den Beteiligten immer selbst organisiert. Es soll in erster Linie dem Austausch zwischen den DV-Abteilungen der einzelnen Institute dienen und praktische Hilfen bzw. Impulse für die tägliche Arbeit geben. Wir glauben, dass es auch beim DV-Treffen 2006 gelungen ist, das hohe Niveau der Vorgänger-Treffen zu halten und wichtige aktuelle Themen zu behandeln.

Berlin, den 27. August 2007

Klaus Hermann, Jörg Kantel

Web 2.0: Werkzeuge für die Wissenschaft

Jörg Kantel

Max-Planck-Institut für Wissenschaftsgeschichte, Berlin

Gebt uns Tools! (Jürgen Renn)¹

Zusammenfassung

Um den Begriff Web 2.0 ist ein gigantischer Buzz entstanden. Dabei existieren viele der Methoden, die sich hinter dem Begriff verbergen, schon seit geraumer Zeit. In diesem Artikel, der eine Ausarbeitung des am 23. DV-Treffen der MPG am 15. November 2006 gehaltenen Workshops ist, möchte ich die wichtigsten dieser Methoden vorstellen, um einmal einen Blick hinter den Hype zu werfen. Dabei werde ich auf Begriffe wie RSS, AJAX, Mashup, Social Software etc. eingehen und die zugrunde liegenden Werkzeuge vorstellen. Darauf aufbauend wird ein Entwurf einer Peer-2-Peer-Web-2.0-

-
1. Im Wortlaut: „Software-instruments for the management of research information as they are needed by the institutes such as project-servers, digital archives, tool-boxes, data-mining tools and the like are not being developed other than in exceptional cases.“ [[Renn 2000] Jürgen Renn: Challenges of the Information Revolution for the Max Planck Society, Berlin (MPIWG Preprint 151) 1999, Seite 6]

Anwendung vorgestellt, die als Modell für wissenschaftliche, kollaborative Arbeitsumgebungen dienen kann.

1. Web 2.0: Buzzword oder Hype oder doch etwas Vernünftiges?

Web 2.0 ist die Summe der Bemühungen, das Web kommunikations- und benutzerfreundlicher als bisher zu gestalten. Es gibt eigentlich nichts wirklich Neues, die Bausteine des Web 2.0 sind seit Jahren bekannt². Allerdings erlauben erst die derzeit möglichen Bandbreiten, diese Techniken flächendeckend zu kombinieren und einzusetzen.

Es gibt keine echte Definition des Web 2.0. In der Regel wird darunter eine Kombination verschiedener Webservice-Anbieter wie z. B. den Photo-Dienstleister Flickr aber auch die Business-Community openBC mit Techniken wie AJAX³ und CSS 2.0 verstanden, die einmal

- verschiedene Dienste und Anwendungen zur Community-Bildung bieten, zum Beispiel auf spezialisierten Servern gemeinsamen Interessen nachzugehen oder über gleiche Tags sich zu verständigen, wer ähnliche Interessen besitzt und
- die Anwendungen im Browser „Desktop“-ähnlicher machen, zum Beispiel Änderungen auf der Seite anzeigen, ohne daß die Seite – wie sonst üblich – im Browser neu geladen werden muss.

Ich werde im Folgenden erst einmal versuchen, diese Elemente vorzustellen, um einen Überblick zu geben, was unter Web 2.0 eigentlich alles verstanden wird oder verstanden werden kann, um später daraus eine Peer-2-Peer-Arbeitsumgebung für kollaboratives Arbeiten im und mit dem Web zu entwickeln.

2. Web 2.0 ist eigentlich nur ein Buzzword, das versucht, zwei unterschiedliche Dinge unter einen Hut zu bringen, der sich dann gut „verkaufen“ lässt. Aber die Techniken sind da und sie werden auch genutzt.

3. AJAX ist eine Abkürzung für *Asynchronous Javascript and XML*. Es bezeichnet ein Konzept der Datenübertragung zwischen einem Server und dem Browser, welches es ermöglicht, dass die HTML-Seite nicht mit jeder HTTP-Protokoll-Anfrage komplett neu geladen werden muss. Das eigentliche Novum besteht in der Tatsache, dass nur gewisse Teile einer HTML-Seite sukzessiv bei Bedarf nachgeladen werden.

2. Personal Publishing: Weblogs und Wikis

2.1 Weblogs

Weblogs haben in den letzten zwei, drei Jahren eine enorme Popularität erreicht. Zwar hinkt Deutschland gegenüber den USA, was die Zahl der aktiven Nutzer betrifft, noch weit hinterher, doch spätestens seit der „Jamba-Affäre“⁴ um den Spreeblick-Blogger Johnny Häusler sind sie auch in das Blickfeld der Medien gerückt.

-
4. Johnny Häusler hatte in einem Bericht in seinem Blog die fragwürdige Praxis von Jamba ironisch glossiert, den Käufern statt eines Klingeltons gleich ein ganzes Klingelton-Abo anzudrehen. Kurz darauf gab es zu diesem Beitrag in seinem Blog viele Kommentare, die diese Praxis von Jamba verteidigten. Ein Blick in die IP-Adressen der Kommentatoren zeigte, dass diese Verteidiger alle von einer einzigen Adresse kamen, der Netzwerkadresse von Jamba selber. Hier versuchten also Jamba-Mitarbeiter anonym, die Meinung für sich zu beeinflussen. Nachdem dies ruchbar wurde, gab es ein enormes Rauschen in der Blogosphäre und bald darauf auch im „normalen“ Blätterwald. Diese Geschichte wird heute immer wieder als Beispiel für den PR-GAU benutzt, der einem in der Blogosphäre widerfahren kann.



Abb. 1: Screenshot des Schockwellenreiters, (m)eines „klassischen“ Weblogs (www.schockwellenreiter.de). Es sind kurze Beiträge, die um einzelne oder auch mehrere Links kreisen, ergänzt durch Bilder, Filme, MP3-Links. Es besitzt das klassische, dreispaltige Layout. Unter den einzelnen Beiträgen stehen bei Bedarf Tags, darunter sieht man Permalink, Kommentare, Trackbacks (wird weiter unten erklärt) und – falls notwendig – auch Links zu entsprechenden Kategorien (Schlagworten). Die linke Spalte enthält in der Hauptsache Werbung und Gimmicks, in der rechten Spalte finden sich Links zu den Archiven, Kategorienseiten und anderen Webseiten. Hier findet man typischerweise auch die Blogroll, Links zu den Weblogs, die der Weblogbetreiber selber liest und/oder empfiehlt.

2.1.1 Was sind eigentlich Weblogs?

Weblogs sind erst einmal Webseiten, die regelmäßig aktualisiert werden. Dabei sind sie in der Regel in umgekehrt chronologischer Reihenfolge sortiert, das heißt, das Neueste steht immer oben, ältere Beiträge landen nach einer Schonfrist im ebenfalls zugänglichen Archiv. In der Regel werden Weblogs mit Hilfe einer speziellen Weblogsoftware erstellt, die es ermöglicht, die Seiten zu pflegen, ohne große Kenntnisse von der darunterliegenden Technik und den Beschreibungs- und Programmiersprachen des Web (HTML, XHTML, CSS, JavaScript etc.) zu haben.

Ursprünglich entstanden Weblogs als kommentierte Linksammlung, die die täglichen Streifzüge durch das Internet dokumentierte. Daher die hohe Bedeutung, die in Weblogs den *Links* beigemessen wird. Daneben bemächtigte sich aber schnell auch eine Gruppe der Weblogsoftware, die schon länger, aber eher unbeachtet, ein Leben im Web führte: die der Tagebuchschreiber. Denn auch sie profitierten von der neuen Weblogsoftware. Hatten sie bisher ihre Homepages mit den Tagebüchern liebevoll aber aufwändig selber gestaltet, verlangte jede Änderung mühevoll Herumfummeln und Beherrschen des darunterliegenden HTML, erlaubten diese neuen Mini-CMS⁵ nun ein schnelles und unmittelbares Publizieren, genau das Richtige für spontane Tagebuchschreiber.

Heute sind Weblogs, respektive die darunterliegende Weblogsoftware, eher eine Hülle denn ein Genre: Sie erlauben eine Vielzahl von Publikationsformen, vom klassischen Weblog über das Tagebuch bis hin zum Online-Journal oder einer Firmen-Website, die weblogartig über Produktneuerungen berichtet. Aber auch Labortagebücher und Berichte von Forschungsgruppen sind damit leicht und schnell zu realisieren⁶.

2.1.2 Weblogs und Webloghosting

Für die spätere Diskussion ist es wichtig, die unterschiedlichen Methoden, wie Weblogs publiziert werden können, einmal kurz vorzustellen.

Da sind zuerst einmal die **gehosteten Blogs** (Weblog-Dienste). Das ist die einfachste und schnellste Form, mit einem eigenen Weblog an die Öffent-

5. CMS: Content Management System

6. Darüberhinaus kann man die Software sogar zu Zwecken „aufbohren“, an die man eigentlich spontan nicht denkt: Wir nutzen in der EDV-Abteilung des MPIWG eine Weblogsoftware als „Trouble Ticket“ für den Support.

lichkeit zu gehen. Man meldet sich (für ein geringes Entgelt oder gar kostenlos) bei einem Webloghoster an, wählt ein Template aus, gibt seinem Blog einen Namen und kann sofort loslegen. Bekannte Webloghoster sind z. B. Blogger⁷ (gehört heute zu Google), Yahoo! 360⁸ oder Twoday.net⁹

Die zweite und meines Wissens häufigste Form ist das **Blog auf gemietetem Webpace** (Weblog-Software). Hier installiert man eine Weblogsoftware bei einem Internet Service Provider (ISP) der Wahl. Da die Software in der Regel PHP und MySQL verlangt, muss der ISP dieses unterstützen. Die bekanntesten Weblog-Tools für diese Methode sind zum Beispiel *Wordpress* (Open Source) oder *MovableType* (kommerziell). Aber es gibt sehr viele andere und die Grenzen zur ersten Version sind auch fließend¹⁰. Im Idealfall hat man bei dieser Form die volle Kontrolle über sein Blog.

Die ressourcenschonendste Methode ist es, sein Weblog auf einem Desktop, also auf seinem *persönlichen Computer* zu führen und dann **statische Seiten** per FTP oder *rsync* oder ähnlichen Mechanismen auf einen Webserver hochzuladen. Da dieser Server nur noch statisches HTML liefern muss, werden an ihn keine großen Anforderungen gestellt, selbst eine preiswerte (und von manchen Providern sogar kostenlos angebotene) *Web-Visitenkarte* reicht dafür aus.

Die Nachteil der letzten Methode ist natürlich, dass etliche der lieb gewonnenen Spielereien, die mit dynamischen Seiten möglich sind, erst einmal nicht funktionieren. Aber wofür gibt es eigentlich die vielen *Webservices*, die man auch in statische Seiten einbinden kann? So ist meinem Weblog kaum anzumerken, dass es aus statischen Seiten besteht. Kommentare, Trackbacks etc. werden via *Webservices* realisiert.

2.2 Wikis

Ein Wiki ist eine Website, deren Seiten von den Benutzern nicht nur gelesen, sondern auch einfach geschrieben werden können. Der Name stammt von *wikiwiki*, dem hawaiischen Wort für „schnell“. Die einzelnen Seiten und Artikel eines Wikis können auf einfache Weise untereinander verlinkt wer-

7. www.blogger.com

8. 360.yahoo.de

9. www.twoday.net

10. So wird zum Beispiel *MovableType* von der Firma *TypePad* dazu benutzt, Weblogs zu hosten.

den. Im Gegensatz zu Weblogs, die – zumindest in ihrer ursprünglichen Form als Logbuch der Datenreisen – extern ausgerichtet sind, also nach *außen* verlinken, verlinken Wikis aufgrund ihrer Struktur vornehmlich nach *innen*. Das bekannteste Wiki ist sicherlich die Wikipedia¹¹.



Abb. 2: Zettelkasten des historischen Seminars der Universität Heidelberg.

Wikis sind älter als Weblogs und fristeten lange ein Schattendasein im Web. Sie waren eigentlich nur in akademischen Zirkeln und im Bereich der Softwareokumentation zu finden. Erst mit dem Aufkommen der Wikipedia und dem Hype um den Begriff *Social Software* wurden Wikis auch in weiten Kreisen innerhalb des Netzes populär. Und mittlerweile ist die Popularität soweit gewachsen, dass viele Journalisten die Wikipedia meinen, wenn sie „Wiki“ schreiben.

Auch wenn es sie natürlich gibt, sind Wikihoster eher die Ausnahme – im Gegensatz zu Webloghostern. Der Standard ist eigentlich ein Wiki auf gemietetem Webespace. Für diesen Zweck gibt es viele Wiki-Tools, vom eher einfachen DokuWiki¹² bis hin zum MediaWiki¹³, dem Tanker unter den Wikis, mit dem auch die Wikipedia betrieben wird.

11. de.wikipedia.org

12. www.splitbrain.org/projects/dokuwiki

13. www.mediawiki.org

Aber Wikis sind nicht nur für das kollaborative Arbeiten geeignet, sie können auch den Zettelkasten ersetzen und mutieren dann zu einem *Personal Information Management*-Werkzeug (PIM). Neben reinen Desktop-Wikis kann man aber auch jedes andere Wiki auf dem Desktop installieren und unter *localhost* betreiben¹⁴.

Den oben schon erwähnten Zustand, dass Wikis ihrer Struktur nach eher nach *innen* verlinken, versucht man durch sogenannte **Interwiki**-Links zu verändern. Das sind Links, die automatisch zwischen verschiedenen Wikis (auch auf verschiedenen Servern) gesetzt werden. Dazu müssen diese Server natürlich einmal dem Wiki bekannt sein und zweitens – zur Vermeidung von Doppeldeutigkeiten – jedes Wiki mit einer eigenen Namensraumerweiterung versehen werden.

Eine besondere Form der Wikis sind *Blikis*, eine Kombination aus Weblog und Wiki. Blikis versuchen, die Weblog-Funktionalität in ein Wiki zu integrieren. Die bekanntesten Vertreter dieser Klasse sind *Vanilla*¹⁵ und *Snip-Snap*¹⁶, ein in Java am Fraunhofer-Institut für Rechnerarchitektur und Softwaretechnik (FIRST) entwickelter Vanilla-Klon.

3. Werkzeuge (1): Community-Server, Ping, Trackback, RSS

Dave Winer ist einer der Pioniere des Webloggens. Er bot unter anderem auch kostenlose Weblog-Services an. Aber da seine Server von Suchmaschinen und Spidern überlaufen waren, suchte er nach einer Möglichkeit, diesen Verkehr zu minimieren. So war die Idee des *Ping* geboren: Bei jedem Neueintrag setzt die Weblogsoftware einen *Ping*¹⁷ an einen so genannten Community-Server ab. Dieser Community-Server weiß nun,

14. Eine sehr komfortables Desktop-Wiki für MacOS X ist MoinX (moinx.antbear.org), ein One-Click-Installer für das in der Scriptsprache *Python* geschriebene Wiki *Moin Moin*. Da man es nicht nur als *privates* sondern auch als *öffentliches Wiki* betreiben kann, bietet es sich auch als kollaboratives Werkzeug (gemeinsamer Zettelkasten) für (kleinere) Projektgruppen an.

15. www.vanillasite.at Vanilla ist in REBOL, einer ziemlich proprietären Scriptsprache, programmiert.

16. snipsnap.org

17. Der Ping ist ein simples XML-Paket, das den Community-Server informiert, dass in einem Weblog oder auf einer Website ein neuer Beitrag erstellt wurde.

dass es etwas Neues auf der Website gibt und schickt einen Spider los, diese neuen Informationen zu sammeln. Gleichzeitig informiert eine Webseite die *Community*, dass es Neues in diesem Weblog gibt.

Doch wo soll der Community-Server eigentlich suchen? Das Spidern von Webseiten ist ein mühseliges Geschäft, da Layout- und Content-Informationen wild gemischt vorliegen. Hier nutzte Winer eine andere Technik, die er schon früher in einem anderen Zusammenhang für Netscape entwickelt hatte: RSS¹⁸. Dies ist eine spezielle XML-Datei, die ursprünglich nur die Überschriften der Beiträge einer Website enthielt, heute aber entweder auch Zusammenfassungen der Beiträge oder gar den Volltext der Beiträge selber. RSS ist *Inhalt pur*, jede Layout-Information der Webseiten ist eliminiert. Das macht RSS speziell für Spider und andere Bots gut lesbar und zur Weiterverarbeitung geeignet.

Winers Idee war es nun, Weblogs neben den HTML-Seiten auch RSS-Seiten (automatisch) publizieren zu lassen und nur die RSS-Seiten von seinem Spider abzufragen. Dies – und die Idee des RSS selber – erwies sich als ungemein erfolgreich. RSS hat heute die begrenzte Welt der Weblogs längst verlassen und dient überall dort im Web als Syndication-Format, wo regelmäßige Updates stattfinden und man die Welt über diese Updates informieren möchte¹⁹.

Ben und Mena Trott, die Entwickler der populären Weblogsoftware *MovableType* haben die Idee des Pings konsequent weiterentwickelt. Schon früh boten Weblogs die Möglichkeit zu kommentieren, doch einige – wie zum Beispiel Winer – stehen dieser Möglichkeit skeptisch gegenüber: Wenn ich einen Weblogbeitrag kommentiere, sollte ich dies nicht konsequenterweise in *meinem* Weblog tun? Die Idee der Trotts war es nun, wenn ich einen Beitrag eines fremden Weblogs in meinem Weblog kommentiere, an dieses Weblog ebenfalls einen Ping abzusetzen, der es darauf hinweist, dass sein Beitrag kommentiert wurde. Der Trackbackping oder kurz *Trackback*²⁰ war geboren.

War es noch Winers Idee, dass der Community-Server die Updates *aller* Weblogs auflisten sollte, entstanden schnell spezialisierte Community-Ser-

18. RSS steht meist für *Rich Side Summary* oder *Really Simple Syndication*, doch darüber streiten sich die Experten noch. Uns soll es egal sein, Hauptsache, wir wissen, wie und das es funktioniert.

19. Vgl. [[Kantel 2007] Jörg Kantel: RSS & Atom kurz und gut, Köln (O'Reilly) 2007].

ver, die nur noch die Updates einer speziellen Community (meist eines einzelnen Weblog-Hosters) listeten. Im deutschsprachigen Raum sammelten sich die ersten in der Antville-Community²¹, eines der ersten europäischen Weblog-Dienste, der dazu auch noch nichtkommerziell ausgerichtet war²². Dabei wurde die Community-Bildung zusätzlich dadurch angeregt, daß nur Mitglieder der Antville-Community in Antville-Blogs kommentieren durften. Dies bildet zwar einerseits einen relativ zuverlässigen Schutz gegen Kommentarspam, ist aber auf der anderen Seite eine Ausgrenzung anderer Webteilnehmer und daher wird diese Form der Community bis heute kontrovers diskutiert. Hier verläuft dann auch irgendwo die Grenze zwischen Weblog und (Diskussions-)Forum. Speziell in diesen geschlossenen Communities gibt es oft Beiträge mit Dutzenden oder gar Hunderten Kommentaren. Hier stehen nicht mehr der Beitrag und sein Autor im Vordergrund, sondern die Diskussion²³. Der Übergang zu Foren ist hier fließend. Aber auch den umgekehrten Fall habe ich schon gesehen: Eine Forensoftware, die zu einem Weblog aufgebohrt wurde.

4. Ajax ist (k)ein Scheuerpulver

Herkömmliche Webanwendungen sind *stateless*, das heißt, der Server, der eine Seite an einen Client ausliefert, hat *kein Gedächtnis*, er weiß nichts von vorhergegangenen Anfragen des gleichen Clients. Darüberhinaus verlangt jede neue Anfrage an einen Server, dass eine Seite komplett neu geladen werden muss, auch wenn zum Beispiel nur ein Satz verändert wurde. Hier

20. Das Trackback-Protokoll ist ein wenig schlampig formuliert und meistens auch implementiert. Daher ist es etwas in Verruf geraten. Dies ist schade, denn es bietet auch die Möglichkeit, z. B. in wissenschaftlichen (Online-)Arbeiten einen Feedback über das **Zitiert-werden** zu geben. Eine „verbesserte“ Form des Trackbacks, das Pingback, hat sich nicht durchgesetzt.

21. www.antville.org

22. Antville ist eigentlich bis heute das „Hobby“-Projekt einiger Wiener Freaks.

23. Dieses Phänomen ist aber nicht ausschließlich an Communities wie die Antville-Community gebunden. Auch der *Spreblick*, das populäre Weblog von Johnny Häusler, ähnelt mehr und mehr einem Forum, was sicher auch an der Diskussionsfreudigkeit von Johnny selber liegt. Und einige *Blawgs*, Weblogs von Rechtsanwälten, sind von spezialisierten Jura-Foren ebenfalls kaum noch zu unterscheiden.

setzt *AJAX* ein: AJAX-Anwendungen verändern mit Hilfe des *Document Object Models* (DOM) und JavaScript den Inhalt einer bestehenden Webseite ohne einen vollständigen Reload der Seite zu erzwingen. Darüberhinaus sind AJAX-Anwendungen *asynchron*, das heißt, während eine Anfrage an einen Server läuft, kann im gleichen Browserfenster weitergearbeitet werden. Das ist das, wofür AJAX steht: Asynchronous Javascript And XML.

Damit wird versucht, Applikationen im Web dem Verhalten von Applikationen auf dem Desktop anzupassen, denken wir zum Beispiel an eine Textverarbeitung, die ein fett markiertes Wort auch sofort als fett markiert anzeigt und nicht erst, nachdem man einen *Submit Button* gedrückt hat. Man kann mittlerweile mit Hilfe von AJAX faszinierende Dinge im Browser erledigen, aber es gibt auch einige Nachteile dieser Technik. Die wichtigsten sind:

AJAX benötigt zwingend **JavaScript**. Menschen, die aus Sicherheitsgründen JavaScript in ihrem Browser ausgeschaltet haben oder die aus irgendwelchen Gründen auf alternative Clients, die kein JavaScript beherrschen ausweichen (müssen) – man denke an Braille-Geräte oder Browser für Sehbehinderte, die den Inhalt des Browserfenster *vorlesen* – sind von AJAX-Anwendungen ausgeschlossen.

Das *Document Object Model* ist in unterschiedlichen Implementierungen je nach Browser vorhanden. Das heißt, die nach dem (verdienten!) Ende von DHTML schon totgeglaubten **Browserweichen** feiern fröhliche Urständ. Und es gibt wieder Webseiten, die z. B. im Safari, Apples hauseigenem Browser, nicht laufen oder die schlicht und einfach nur in Browsern der Mozilla-Familie funktionieren.

Die Navigation mit Hilfe des **Back Buttons** läuft nicht mehr, wie der Nutzer es gewohnt ist. In der Regel löscht ein *zurück* nicht mehr den letzten Eintrag, sondern setzt die Seite in den Ursprungszustand, bevor die AJAX-Anwendung startete, zurück. Dieses Verhalten ist vom Benutzer oft nicht vorhersehbar – er weiß nicht, ob gerade eine AJAX- oder eine *normale* HTML-Anwendung läuft.

Damit verwandt ist das Problem der **Deep Links**: AJAX-Anwendungen können (ohne programmiertechnische Klimmzüge) nur auf ihren Ursprungszustand verlinkt werden.

Dass AJAX Schwierigkeiten bei der Gestaltung von **barrierefreien** Webseiten hat, wurde weiter oben schon angesprochen. Hier wird in der Literatur als Ausweg fast immer empfohlen, eine zweite, AJAX- und JavaScript-freie Site parallel bereitzustellen. Dies ist natürlich ein beträchtlicher Mehrauf-

wand und es ist immer zu prüfen, ob die Vorzüge von AJAX diesen Mehraufwand rechtfertigen.

Fazit: AJAX ist eine faszinierende Technologie, die aber wegen der genannten Probleme nur mit Bedacht und Umsicht eingesetzt werden sollte.

5. Webservices & Mashups

Webservices sind verteilte Funktionen oder Prozeduren, die über Web-Techniken – in der Regel via http – erreichbar sind. Der Prozedur-Aufruf erfolgt fast immer über ein *GET* oder *POST* und zurückgeliefert wird meist XML, manchmal aber auch Text pur oder JavaScript. Diese Antworten werden dann via Scriptsprachen oder AJAX in eine Webseite eingebunden. Google war einer der Pioniere der Webservices, populär wurden sie allerdings vor allem durch kommerzielle Anbieter wie Amazon oder Ebay, die es den Site-Betreibern zum Beispiel ermöglichten, einzelne Produkte auf ihren Webseiten einzubinden.

Die gängigsten Webservice-Methoden sind im Einzelnen:

- **XML-RPC** ist eine der ältesten Methoden, die von *Dave Winer* entwickelt wurde, weil ihm die Arbeit an SOAP (in dessen Kommission Winer auch saß) zu lange dauerte. XML-RPC ist ein recht simples Protokoll, das als Anfrage wie auch als Antwort XML-Pakete erwartet. Wegen seiner Einfachheit ist XML-RPC auch heute noch sehr beliebt und bei vielen Webservices im Einsatz.
- **SOAP**, das *Simple Object Application Protocol*, meist zusammen mit *WSDL*, der *WebService Description Language*, verwendet, ist ebenfalls ein auf XML basierendes RPC-Protokoll, das bedeutend umfangreicher und damit auch komplizierter als XML-RPC ist. *SOAP* wird in der Hauptsache in Enterprise-Anwendungen eingesetzt, aber auch das Google-API war bis vor kurzem eine reine *SOAP*-Anwendung²⁴.
- Das dritte im Bunde der Webservice-Protokolle ist **REST**, der *Representational State Transfer*. Hier werden Anfragen via PUT, GET, POST oder DELETE an URIs verschickt, die die Parameterübergabe regeln und zurück kommt eine Antwort entweder als XML (die Regel) oder als HTML, manchmal aber auch als Text pur oder in einem beliebigen anderen Format – *REST* macht da keine Vorschriften.

24. Mittlerweile hat Google das SOAP-API aufgekündigt und setzt für die Zukunft komplett auf AJAX.

- Natürlich sind Webservices auch via **JavaScript** und/oder **AJAX** zu realisieren. Hier ist das Antwortformat oft **JSON**, die *JavaScript Object Notation*, die natürlich von JavaScript direkt ohne aufwendiges Parsen gelesen werden kann.

JSON ist mittlerweile so erfolgreich, dass auch Anwendungen in Java, PHP, Perl und Python existieren, die dieses Format nutzen.

- Und man kann auch (einfache) Webservices mithilfe von **RSS** realisieren. Diese Möglichkeit wird viel von Werbetreibenden genützt, die damit immer aktuelle Anzeigen in die Partnerseiten einbinden.

Webservices und die dazugehörigen APIs (*Application Programming Interfaces*) werden heute von einer Vielzahl von Webdiensten angeboten und bilden die Grundlage für **Mashups**, die aus diversen Webservices (und womöglich wieder mit Hilfe von AJAX) eine Anwendung zusammenbauen.

Webservices und Mashups können auch für *alternative Clients* (z. B. Google Earth²⁵, Yahoo! -Widgets²⁶ (vormals Konfabulator) oder Apples Dashboard²⁷) entwickelt werden.

6. Social Software

Community-Server sind heute mehr als reine Update- und Inhaltsverzeichnisse von Weblogs einer Community. *Technorati*²⁸ zum Beispiel ist so etwas wie ein Ranking-Verzeichnis (welches Weblog ist wie populär, wobei die Popularität nach der Zahl der Verlinkungen entschieden wird), eine Weblog-Suchmaschine und eine Art verbesserter Trackback-Server (welches Weblog linkt auf mein Weblog). Technorati basiert auf dem Prinzip des *Pings*, um von Technorati indiziert zu werden, muss man Technorati anpingen.

Daneben indiziert Technorati spezialisierte Tags und kann aufzeigen, welche Weblogbeiträge welche Tags verwendet haben. Man kann danach suchen oder sich Tagwolken anzeigen lassen.

Weitere populäre, aber nicht die einzigen Webservices, die Tags verwenden, sind die Photo-Community Flickr²⁹, die kürzlich von Yahoo! gekauft wurde,

25. earth.google.com

26. widgets.yahoo.com

27. www.apple.com/de/macosx/features/dashboard/

28. www.technorati.com

und der Bookmark-Online-Dienst del.icio.us³⁰ – ebenfalls im Dezember 2005 von Yahoo! aufgekauft. Beides sind Server-Dienste, das heißt die Tags werden nicht via Ping und RSS eingesammelt, sondern die Links und Photos werden direkt auf dem Server getagged.

Im Videobereich ist die Frage nach dem *Platzhirschen* unter den verschiedenen Anbietern noch nicht endgültig entschieden, allerdings wächst *YouTube*³¹ ziemlich schnell und nachdem Google diesen Videodienst für ca. 1,6 Mrd. Dollar gekauft hat, scheint auch hier das Rennen gelaufen zu sein.

Populär und geradezu mit dem Begriff Web 2.0 verbunden sind aber auch die so genannten *Social Networks*, Anwendungen, deren Hauptzweck es ist, die Nutzer miteinander zu vernetzen und miteinander kommunizieren zu lassen. Die Bandbreite ist vielfältig: Da gibt es reine Business-Portale, wie das zu *XING*³² mutierte, ehemalige *openBC*, Zielgruppenportale wie das kürzlich etwas in Verruf geratene und von Holtzbrinck gekaufte *studiVZ*³³, das sich an Studenten wendet, eher unspezifische Teenager-Portale wie das von Mur-

29. www.flickr.com

30. del.icio.us

31. www.youtube.com

32. www.xing.com

33. www.studivz.net

doch gekaufte *mySpace*³⁴, aber auch so etwas wie das von Google betriebene *Orkut*³⁵, das ich eher als Kontaktbörse einsortieren würde.

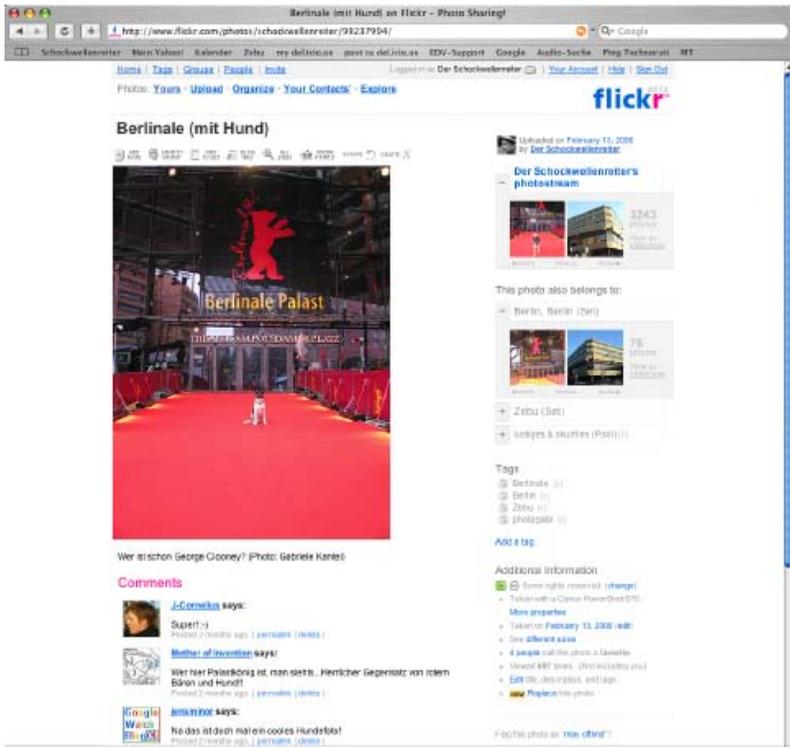


Abb. 3: Screenshot einer typischen *Web 2.0*-Umgebung am Beispiel der Photo-Community „Flickr“: Rechts neben dem großen Photo sieht man, dass dieses auch noch diversen Sets und Gruppen zugeordnet ist. Darunter sind die *Tags* zu erkennen. Die jeweiligen Sets und Gruppen kann man mit einem Klick auf das Plus- (+) oder Minus-Symbol (-) auf- oder zuklappen. Dies, wie auch das Hinzufügen von weiteren Tags ist via *AJAX* realisiert. Unter dem Bild sieht

34. www.myspace.com

35. www.orkut.com

man Kommentare, neben den Gruppen und *Tags* ein weiteres *Community*-förderndes Feature von Flickr. (Photo: Gabriele Kantel)

7. Das Web ist multimedial: Photos, Pod- und Vodcasts

Im Gegensatz zu den vorherigen, rein textbasierten Anwendungen des Internet (mail, newsgroups, gopher etc.) war das World Wide Web von Anfang an für die Präsentation multimedialer Inhalte ausgelegt. Schon die ersten Seiten zeigten Bilder, früh kamen Töne und Filme dazu. Doch während die Integration von Bildern in Webseiten von Anfang an genormt war, taten sich die Macher des W3C seltsamerweise mit anderen Inhalten schwer. Das führte dazu, dass die Browserhersteller den nicht standardkonformen `<embed>`-Tag einführten, der bis heute auf vielen Webseiten sein (Un-)Wesen treibt. Der vom W3C vorgeschlagene und viel umfassendere `<object>`-Tag wurde dann nur sehr langsam in die Browser implementiert, so dass der Web-Designer bis heute in der Regel zur Sicherheit mit einer Kombination beider Tags arbeitet, wenn es gilt, multimediale Inhalte einzubinden.

Weiter herrscht Unklarheit darüber, *welche* Formate die Browser verstehen sollten. Während es bei Pixelbildern noch relativ einfach ist, alle Browser verstanden von Anfang an das GIF- und das JPEG-Format, später kam aus patentrechtlichen Gründen noch das (offene) PNG-Format hinzu, herrscht bei Audio- und Video-Formaten fröhlicher Wildwuchs: Microsofts eigene Formate verstand (natürlich) der Microsoft-eigene Browser Internet Explorer, und da er lange Jahre so etwas wie ein 99-Prozent-Browser war, was die Verbreitung anging, glaubten viele, dass das, was der Internet Explorer versteht, auch vom Rest der Browserwelt verstanden wird. Und wenn nicht, war es ihnen egal.

Im Audio-Bereich scheint sich mittlerweile das MP3-Format gegenüber allen anderen Formaten als Quasi-Standard durchzusetzen. Diese Entwicklung birgt jedoch eine Gefahr, da das MP3-Format nicht frei ist, sondern „jemanden gehört“, der im Prinzip damit machen kann, was er will, also auch die zukünftige Nutzung verbieten. Das Ogg-Vorbis-Format ist eine freie Alternative mit zur Zeit leider noch begrenzter Verbreitung.

Weiterhin fröhlicher Wildwuchs herrscht hingegen bei den Video-Formaten: Zum einen streiten die drei „Großen“ Real, Microsoft und Apple mit ihren jeweils eigenen Formaten um die Marktführung, zum anderen bedeutet es noch lange nicht, dass, wenn man den richtigen Player für das richtige Format gefunden hat, der Film auch abspielbar ist; zu viele Codices können den Erfolg verhindern, da die Formate eigentlich nur Container sind, die diverse

Multimedia-Inhalte enthalten können. Hier sollte der Multimedia-Produzent vorher ausgiebig testen, bevor er sein Produkt ins Netz stellt.

7.0.1 Exkurs: Flash und die Alternativen dazu

Und Flash? Flash ist ein proprietäres Vektorgraphik- und Animationsformat, das von Macromedia speziell für den Einsatz im World Wide Web entworfen wurde. Und da die Normierungsbemühungen des W3C nur sehr langsam vorangingen, Macromedia aber die Consumer-Betriebssysteme Windows und Macintosh schon sehr früh mit Plug-Ins versorgte, die es den Browsern erlaubten, Flash-Filme *im* Browser abzuspielen, wurde Flash schnell so etwas wie ein Quasi-Standard für Animationen. Abgesehen davon, daß damit auch sehr viel Unsinn angestellt wurde, – jeder kennt die nervenden Intros auf Webseiten, die höchstens beim ersten Betrachten witzig sind – ist es für bestimmte Anwendungen (Spieleentwicklung!) ein nützliches Format, für das aber bestimmte Einschränkungen gelten:

- Flash-Dateien sind binär: Das heißt, nur der (im Browser integrierte) Flash-Player kann sie interpretieren. Insbesondere Suchmaschinen, Lesegeräte für Blinde und anderes können mit Flash-Dateien nichts anfangen.
- Zwar ist das Flash-Format offengelegt worden, so dass man mittlerweile auch Flash-Filme erstellen kann, *ohne* Macromedias teure Entwicklungsumgebung zu benutzen, aber das Flash-Format gehört Macromedia, resp. seit der Übernahme durch Adobe gehört es Adobe. Und Formate, die jemanden gehören, sind immer von einer ungewissen Zukunft betroffen.³⁶ So gab es zum Beispiel für einige Zeit, obwohl vorhanden, keinen Flash-Player in den FreeBSD-Distributionen. Die Lizenzbedingungen von Adobe verhinderten dies.

Für die Erstellung interaktiver Filmchen und Spiele ist Flash aber trotz allem sicher geeignet. Und mit der ECMA-konformen internen Scriptsprache *ActionScript*, einem JavaScript-Dialekt, ist Macromedia auch einen Schritt weiter in Richtung Standardkonformität gegangen. Aber es gibt durchaus eine standardkonforme und offene Alternative:

36. Wie wenig sich Adobe da um einen breiten Benutzerkreis schert, zeigt das Beispiel von *FrameMaker*. Obwohl Frame nach Meinung vieler für technisches Publizieren das beste Programm war, wurde es nach dem Kauf von Adobe kaltgestellt. Man wollte dort wohl eher sein eigenes Produkt *InDesign* bevorzugen.

SVG ist ein vom W3C genormtes offenes XML-basiertes Format für Vektorgraphiken und -Animationen. Wie zum Beispiel die Webseiten des Statistischen Bundesamtes zeigen, kann man damit alles anstellen, was man auch mit Flash anstellen kann. „Programmiert“ wird SVG in JavaScript.

Gegenüber Flash hat SVG einige Vorteile:

- SVG ist ein XML-basiertes und damit ein Textformat. Das heißt, es kann sowohl von Suchmaschinen durchsucht als auch zum Beispiel von Braille-Lesegeräten gelesen und interpretiert werden. Und zum Erstellen von SVG-Dokumenten reicht – im Prinzip – ein einfacher Texteditor.
- SVG ist ein offener Standard!

Allerdings gibt es bis heute noch einige Nachteile:

- Die SVG-Unterstützung der Browserhersteller läuft nur sehr zögerlich an. Man ist in der Regel auf das Plug-In von Adobe angewiesen, wenn man SVG-Dokumente im Browser betrachten will.³⁷
- Ähnlich sieht es mit den Werkzeugen zur Erstellung von SVG-Dokumenten aus. Während statische Zeichnungen mittlerweile gut unterstützt werden (fast alle Vektorgraphik- und CAD-Programme bieten mittlerweile einen Export nach SVG an), fehlt es für Animationen an einem komfortablen Autorentool wie Flash.

SMIL ist eine weitere vom W3C standardisierte offene XML-Sprache, die speziell als Container für Multimedia-Inhalte im Web gedacht ist. Mithilfe von SMIL sind interaktive und verlinkte Filme mit einer Zeitleiste möglich. SMIL ist mit SVG verwandt, daher gilt das oben Gesagte auch für SMIL. SMIL ist auch deshalb wichtig und sollte bei der Überlegung der Formatnutzung immer mit einbezogen werden, weil es als offenes Format nicht wie Flash von der Gutwilligkeit des Rechteinhabers abhängig ist und sich daher auch speziell (aber nicht nur) für die Langzeitarchivierung eignet³⁸.

37. Allerdings wurde in der letzten Zeit die Unterstützung von SVG forciert. Von Opera, Firefox und Safari gibt es mittlerweile (Beta-)Versionen, die SVG *native* unterstützen. Das ist auch bitter nötig, da Adobe die Unterstützung des SVG-Plugins aufgekündigt hat.

38. Hierbei ist aber zu beachten, dass SMIL ebenfalls nur einen Container für Multimedia-Inhalte bereitstellt. Für die Abspelbarkeit und – vor allem – die Langzeitarchivierung ist daher die Frage der eingebetteten Formate (und Codices) noch offen.

7.1 Turn Your Radio On oder Podcast ist (nicht nur) Internetradio

Podcasting ist die aktuelle Sau, die gerade durch die Medien getrieben wird. Der Begriff setzt sich zusammen aus iPod (Apples portabler Multimedia-Player) und Broadcasting (Rundfunk). Gemeint ist das Anbieten von Audio-Dateien (im MP3-Format) im Netz, die heruntergeladen werden können und die man sich dann in seinem MP3-Player auch unterwegs anhören kann. Podcasts sind damit ein sowohl synchrones als auch ein asynchrones Medium.

Podcasts sind an sich nichts Neues. Schon vor Jahren führte Dave Winer für seine Weblog-Software *Radio UserLand*³⁹ eine Erweiterung im RSS-Feed ein, die Attachments, also Dateien, die heruntergeladen werden können, in den Feed einband. Als *Grateful Dead*-Fan dachte Winer damals eher an Musikdateien, die man in seinem Blog anbot⁴⁰. Und Töne aufnehmen konnte man mit Computern ebenfalls schon lange. Aber wohl erst mit dem Hype um Apples iPod wurde das Aufnehmen und Abspielen von Audio-Dateien über das Netz richtig populär.

Wer im Netz also *in* sein will, der liest vor. Und man braucht dafür erstaunlich wenig Technik. Im einfachsten Fall reicht das eingebaute Mikrofon und die mitgelieferte Recording-Software des Rechners. Größer sind die rechtlichen Probleme: Wenn man „Radio im Internet“ machen will, möchte man auch Musik abspielen. Aber während die GEMA immerhin ein – wenn auch überteuertes – Modell anbietet, mit dem man sein Podcast legalisieren kann, lebt die Vertretung der aufführenden Musiker, die GVL, noch völlig hinter dem Mond. Man darf zwar nur Podcasts abspielen, wenn man entweder eine Genehmigung der GVL oder aber die Genehmigung der aufführenden Musiker selber besitzt, aber auf Nachfrage⁴¹ erklärt sich die GVL nicht für zuständig. Man muss also für seine Podcasts jeden einzelnen aufführenden Musiker anschreiben und um Genehmigung bitten – eine absurde Situation.

39. radio.userland.com

40. Radio UserLand begann seine Karriere ja auch nicht als Weblog-Software, sondern als Software zum Abhören von MP3s. Erst nach einem der berechtigten Winerschen *Cornerturns* wurde daraus eine Weblogsoftware.

41. Spreeblick Weblog vom 14. Juni 2005 (<http://www.spreeblick.com/2005/07/14/podcasts-radio/>), zuletzt besucht am 10. März 2006.

So bestehen Podcasts heute in der Regel aus freier Musik und aus vorgelesenen Texten und sind somit weit entfernt von dem Potenzial, das in ihnen stecken könnte. Man denke nur an die Musik, die heute im Dudelfunk nicht mehr gespielt wird, an alte Kabarett-Aufnahmen etc.⁴² Hier liegen noch eine Menge an Chancen und Möglichkeiten brach.

7.2 Alles bewegt sich und alles ist bunt – Video im Netz

Videos im Netz sind heute dank größerer Bandbreite weit entfernt von dem früher belächelten „Video auf einer Briefmarke“-Format. Das Netz ist auf dem besten Wege, ein neuer Distributionskanal für bewegte Bilder zu werden. Und das nicht nur im kommerziellen Bereich. Video-Blogging (auch *Vlogging* oder *Vodcasting* genannt) ist ein stark wachsendes Subgenre und auch Videochats und Videotelephonie werden dank iChat, iSight und ähnlichen Werkzeugen immer populärer.

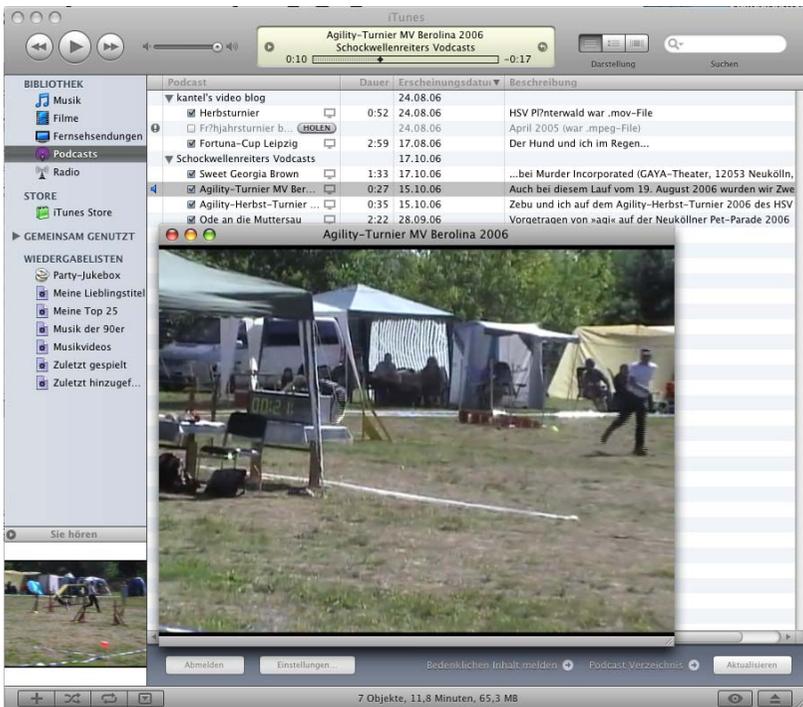


Abb. 4: Screenshot eines Videocasts in iTunes. Dieser lässt sich auch auf dafür geeignete Player herunterladen und dann dort abspielen – auch wenn man *offline* ist.

42. Ich träume zum Beispiel davon, eine Geschichte des *Deutsch-Folk* als Podcast zu produzieren, aber wie soll ich das rechtlich realisieren?

Apple hat als einer der ersten großen Player diesen Trend sehr schnell erkannt und Video-fähige iPods auf den Markt geworfen. Und auch der MP3-Player von Apple (iTunes), den es nicht für für Macintosh-Computer, sondern auch für Windows gibt, kann seit geraumer Zeit nicht nur Podcasts, sondern auch Videocasts abspielen. Diese sind – ähnlich wie Podcasts – entweder über spezialisierte RSS-Feeds oder über den iTunes-Music-Store (ein Online-Store von Apple) zu abonnieren und können dann natürlich auch auf den iPod übertragen werden.

Das bedeutet, dass man sich morgens seine Nachrichten auf den iPod laden und sie dann zum Beispiel in der U-Bahn anhören **und** anschauen kann. Ein ganz neues Einsatzgebiet für das morgendliche Frühstückfernsehen...

7.3 Democracy – die DRM-freie Alternative

Pod- und Vodcasting haben sicher durch Apples iPod und den *iTunes (Music) Store (iTMS)* die initiale Zündung erhalten und sind immer noch ein wichtiger Vertriebskanal. Jedoch besitzt dieser einen schwerwiegenden Nachteil: Alle Dateien, die darüber vertrieben werden, sind mit einem DRM⁴³-Wasserzeichen versehen, das das Abspielen auf anderen Geräten wie den iPod und auf anderer Software wie iTunes verhindert. Und das gilt auch ungefragt für eigene Dateien, die man via iTMS vertreiben will. Dagegen und nicht nur dagegen gibt es eine interessante Alternative: **Democracy**⁴⁴, das sich nicht nur als alternativer Vertriebskanal, sondern als die DRM-freie und auf Open Source Software basierende *Zukunft des Internet-TVs* versteht.

Dabei ist *Democracy* auch für die Fragestellung dieses Aufsatzes interessant. Es ist nämlich nicht nur ein alternativer Videoplayer, sondern um *Democracy* herum existiert eine komplette Umgebung, die als Prototyp für eine Peer-2-Peer-Web-2.0-Anwendung betrachtet werden kann.

Mit *Videobomb*⁴⁵ existiert ein *Community-Server*, der neben den üblichen Funktionen wie Anzeige der neuesten von den Nutzern eingestellten Videos, Bereitstellung eines RSS-Feeds etc. auch die Möglichkeit bietet, eigene *Channels* einzurichten, zu verwalten und ebenfalls wieder als RSS-Feed anzubieten. Im Gegensatz zu Videohostern wie *YouTube*, *Google Video*,

43. Digital Right Management

44. www.getdemocracy.org

45. www.videobomb.com

Sevenload oder *MyVideo* kann man allerdings keine Videos dort hochladen, man gibt die URL der Website ein⁴⁶, auf der das Video liegt.

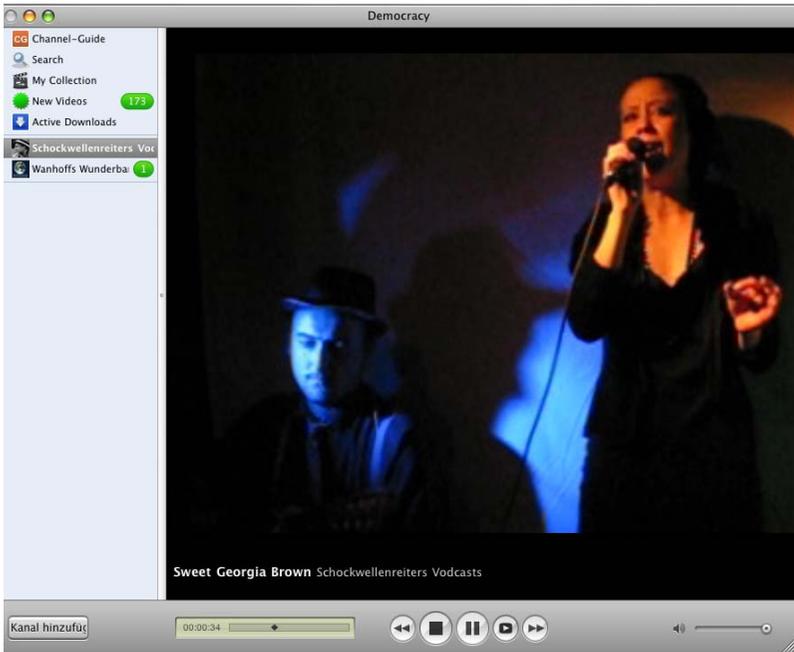


Abb. 5: Democracy – die DRM-freie Zukunft des Internet-Fernsehens?

Darum hat die Community mit der *Broadcast Machine*⁴⁷ eine eigene Software für ein *persönliches YouTube* bereitgestellt, das die Verwaltung eigener, hochgeladener Videos unterstützt, aber auch hier wieder das Einbinden von Videos, die bei anderen Hostern liegen, erlaubt⁴⁸. Die *Broadcast Machine* kann sowohl als persönliches Videolog wie auch als das Videolog und -repository einer Gruppe geführt werden.

46. Es funktioniert auch mit dem bereitgestellten Code zum Einbinden von Videos der großen Videohoster.

47. www.getdemocracy.com/broadcast

48. Zur Zeit macht die Software gerade den Prozess eines kompletten Redesigns durch, da sich die Installation und Wartung des bisherigen Codes doch als etwas problematisch erwiesen hat.

Schließlich gibt es noch den Channel-Guide⁴⁹, der als Alternative zu Apples iTunes designed ist. Hier werden alle *Democracy.org* bekannten Videochannels gelistet und sind ähnlich wie die Vodcasts im iTunes vom Nutzer in *Democracy* abonnierbar⁵⁰.

Democracy ist ein noch junges Experiment, das von der *Participatory Culture Foundation*⁵¹ vorangetrieben wird. Es ist sicher noch nicht perfekt – so fehlt zum Beispiel noch so etwas wie ein *Ping* von der *Broadcast Machine* an den Community-Server (*VideoBomb*) und eine indirekte oder direkte Kommunikation zwischen den einzelnen *Broadcast Machines* ist auch noch nicht implementiert, aber meiner Meinung nach ist das Projekt auf dem richtigen Weg. Ich werde zum Abschluss dieses Aufsatzes noch einmal darauf zurückkommen.

7.4 Moblogging – Bilder, Filme und Töne von überall und sofort

Der Clou in all den oben besprochenen Multimedia-Möglichkeiten des Netzes ist aber ihre unvergleichliche Schnelligkeit, die in der Kombination digitaler Aufnahmegерäte (Photoapparat, Videokamera etc.) mit einer Weblog-Software als MiniCMS liegt. Wir haben schon 2003 ein mehrtägiges, lokales kulturelles Ereignis, die *48 Stunden Neukölln*⁵², mit Digitalkamera und Camcorder begleitet, nach etwa zwei bis drei Stunden „auf der Straße“ haben wir die Ergebnisse zu Hause am Rechner bearbeitet und sofort ins Netz gestellt, d. h. wenige Stunden nach dem Ereignis stand eine Multimedia-Reportage in unserem Neuköllner Blog.

49. Der *channel guide* wird mittlerweile nicht nur von *Democracy*, sondern zum Beispiel auch von *current.tv*, einer Art „kommerziellen, offenen Kanal“ genutzt. (Vgl. [[Pantle 2006] Henrik Pantle: Beyond Broadcast - Rundfunk auf Abruf. Konvergenz von „Broadcast“ und „On Demand“ und die Chancen von User Generated Content, in: Prof. Dr.-Ing Klaus Regensburg (Hrsg.): 2. Tagung Neue Medien und Technologien der Informationsgesellschaft 2006 „Film, Computer und Fernsehen“, Berlin 2006])

50. Es liegt vermutlich in der Natur der Sache eines freien, nichtzensierten Video-Vertriebsweges, dass dort (vor allem, weil es vielfach „hochgeranked“ wurde) sehr viele Videos mit sexueller Darstellung zu finden sind. Trotzdem ist explizit pornographisches Material eher die Ausnahme.

51. www.participatoryculture.org

52. Die Reportage steht immer noch im Netz: <http://www.rollberg.de/achtundvierzigstunden/2003/index.html>

7.4.1 Photo-Handy

Es geht aber noch schneller: Heutige Mobiltelefone entwickeln sich immer mehr zu einer eierlegenden Wollmilchsau: Sie liefern digitale Photos in mittlerweile akzeptabler Qualität, man kann mit ihnen kleine Filmchen aufnehmen und zur Not ersetzt das eingebaute Diktiergerät auch schon einmal den Audio-Recorder.

Moderne Weblogsoftware erlaubt es schon, Multimedienachrichten vom Handy direkt in ein Weblog zu posten, das heißt, man kann mit seinem Handy unmittelbar und nahezu in Echtzeit direkt berichten.⁵³ Dieses *Moblogging* ist in der letzten Zeit unheimlich populär geworden. Und die Handy-Abdeckung bei Jugendlichen liegt nach einer neuen Umfrage bei 100 Prozent. Was dies für Möglichkeiten von (Live-)Reportagen und Ähnlichem bietet, ist noch gar nicht richtig ausgelotet.

8. Werkzeuge (2): Folksonomy – Taggen was das Zeug hält

Jeder kennt das Problem mit den Schlagworten. Sie sollen eindeutig, hierarchisch gegliedert und unverwechselbar sein, um ein vollständiges Wiederfinden aller verschlagworteten Beiträge zu gewährleisten. Dies führt dazu, dass die Diskussion über ein Schlagwortsystem ausufert und die eigentliche Arbeit behindert. Ein Beispiel hierfür ist die nun schon seit Jahren andauernde Diskussion über den *Dublin Core*, die alle bibliothekarischen Feinheiten berücksichtigen will und daher nicht zu Potte kommt.

Tagging dagegen übernimmt einen anderen Ansatz: Jeder versieht seine Beiträge mit den Schlagworten (*Tags*), die er für sinnvoll hält. Diese Tags werden wieder, entweder mit Hilfe eines *Pings* oder es wird – wie z. B. bei *del.icio.us* oder *Flickr* – direkt auf dem Community-Server getaggt.

Nach diesen Tags kann nun wiederum webseitenübergreifend gesucht werden. Zwar findet man sicher bei der Suche über Tags nicht *alle* Seiten, die man finden möchte, aber die Ergebnisse sind in der Regel erstaunlich (semi-)vollständig. Hier spielt sicher das Prinzip der *Selbstorganisation* eine Rolle – wenn ich möchte, dass mein Beitrag wiedergefunden wird, schaue ich vorher nach, welche Tags andere in ähnlichen Beiträgen verwendet

53. Momentan ist diese Möglichkeit meines Wissens bei allen Geräten noch auf Photos und Texte beschränkt, Nokia hat aber schon den nächsten Hack angekündigt, der es erlaubt, auch MP3s und Videos direkt vom Handy ins Netz zu stellen. Denn die Technik dazu ist im Prinzip vorhanden.

haben, aber auch der Umstand, dass man einerseits beliebig viele Tags vergeben kann (so kann ich zum Beispiel einen deutschen Beitrag über Architektur nicht nur mit „Architektur“, sondern auch mit „Architecture“ taggen, um sicherzustellen, dass auch englischsprachige Sucher meinen Beitrag finden) und dass Personen, die an ähnlichen Problemen und Aufgaben arbeiten, ein ähnliches Vokabular benutzen.

Jedenfalls scheinen Tags hinreichend gut zu funktionieren, ohne dass eine Kommission vorher die Art und den Umfang des Tagging-Vokabulars festlegen muss.

Dazu tragen sicher auch die beliebten *Tagwolken* bei, die die populärsten Tags einer Community oder eines Weblogs anzeigen, wobei die Häufigkeit durch unterschiedliche Schriftgrößen angezeigt wird – je häufiger ein Tag verwendet wird, desto größer wird er dargestellt.

9. Web-Mapping

Eine der Killer-Applikationen im Umfeld des Web-2.0-Booms sind sicher die Kartenapplikationen von Google und Yahoo!, die ursprünglich nur als Ergänzung zur jeweiligen lokalen Suche der Portalanbieter gedacht waren: Der Benutzer sollte seine Suche zum Beispiel nach dem nächsten Pizzaladen zur besseren Orientierung auch auf einer Landkarte angezeigt bekommen. Beide Dienste boten aber von Anfang an Schnittstellen an, mit denen man die Karten (kostenlos) auch in eigenen Webanwendungen einbinden konnte. Das war mehr oder weniger die Geburtsstunde der Mashups. Als dann auch noch Satellitenphotos und später sogar hochaufgelöste Luftaufnahmen zu dem Kartenmaterial hinzukam, kannte der Run auf Google Maps und Co. keine Grenzen mehr.

Und schließlich präsentierte Google mit *Google Earth*⁵⁴ einen eigenen, spezialisierten Web-Client, der die Landkarten nicht nur hochaufgelöst darstellen und bei Bedarf aus dem Web nachladen konnte, sondern auch noch über (zuerst rudimentäre) 3D-Fähigkeiten verfügte. Zug um Zug wurde zudem die Qualität des Kartenmaterials und der Luftaufnahmen verbessert⁵⁵. Außerdem wurde mit KML, der *Keyhole Markup Language*⁵⁶, ein auf

54. earth.google.com

55. Es gab aber auch Rückschläge: So wurden zum Beispiel auf Betreiben der britischen Regierung hochaufgelöste Karten aus dem Irak in der Nähe von britischen Militärstützpunkten von Google wieder aus der Datenbasis entfernt und durch (schlechtere) Vorkriegsaufnahmen ersetzt.

GML⁵⁷ basierender XML-Dialekt, eine einfache Möglichkeit präsentiert, eigene „Applikationen“ in *Google Earth* zu entwickeln. Man kann so Overlays definieren und Photos und sogar Filme in *Google Earth* integrieren. Außerdem ist es möglich, 3D-Modelle in die *Google Earth*-Landschaft einzubinden⁵⁸. Google hat dafür extra ein 3D-Programm (*Google Sketchup*) aufgekauft und in der Grundversion kostenlos unters Volk gebracht.

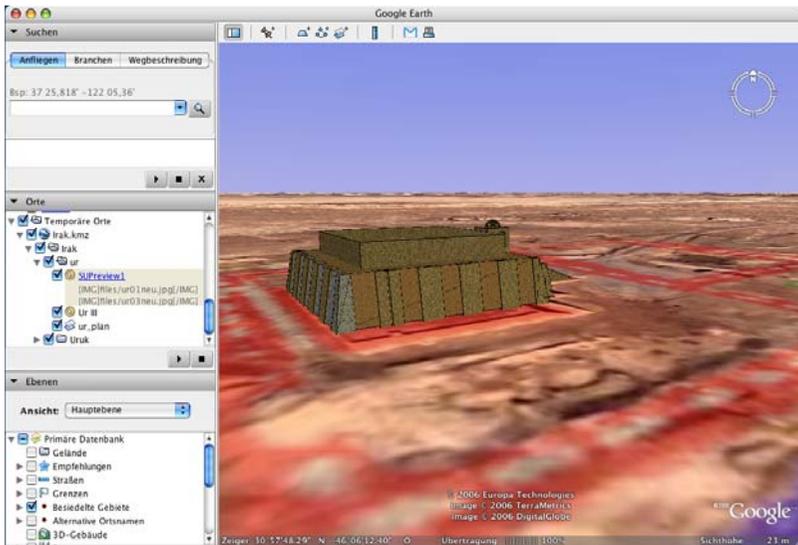


Abb. 6: Mit Hilfe von SketchUp erstelltes 3D-Modell eines Tempels in Ur III, eingebunden in Google Earth, vgl. [[Schröder 2007] Sebastian Schröder: Web 2.0-Techniken für die CDLI (Cuneiform Digital Library Initiative), in diesem Band]

9.1 Exkurs: Google Earth und die Wissenschaft

Welche Möglichkeiten *Google Earth* für die Wissenschaft bietet, hat uns Sebastian Schröder [[Schröder 2007] Sebastian Schröder: Web 2.0-Techni-

-
- 56. Keyhole war die Firma, die Google Earth ursprünglich entwickelt hatte und dann von Google aufgekauft wurde.
 - 57. *Geographic Markup Language*, ein ISO-Standard für geographische Daten.
 - 58. Bis auf die Einbindung von 3D funktioniert dies analog auch mit *Google Maps*.

ken für die CDLI (Cuneiform Digital Library Initiative), in diesem Band] auf dieser Tagung eindrucksvoll demonstriert. Er zeigte am Beispiel der *Cuneiform Digital Library Initiative (CDLI)*⁵⁹, wie man Grabungskarten mit Luftaufnahmen und 3D-Modellen kombinieren, Photos der Grabungsstätten und der Fundstücke hinzufügen und so eine neue Form der Präsentation wissenschaftlicher Ergebnisse in und mit Hilfe von *Google Earth* schaffen kann.

Jedoch darf man nicht vergessen, dass *Google Earth* ein kommerzielles Produkt und nur in der Grundversion frei („frei“ im Sinne von „Freibier“) verfügbar ist. Einmal muss man für komplexere Anwendungen auf eine kostenpflichtige Version zurückgreifen und zum anderen unter Umständen sogar einen eigenen *Google Earth*-Server aufsetzen.

Hier bieten sich unter Umständen freie Alternativen an. So existiert einmal mit dem Projekt *NASA World Wind*⁶⁰ ein Projekt aus der Public Domain an, das *Google Earth* ziemlich ähnlich ist⁶¹ und darüberhinaus nicht nur die Erde, sondern auch Mond und Mars kartographisch erfasst hat. Leider fehlen die hochauflösten Karten, wie Google sie bietet, und der Client ist zur Zeit nur für die Microsoft-Betriebssysteme verfügbar⁶².

Darüberhinaus existieren im Rahmen des *Open GIS Movements* zahlreiche Alternativen aus dem Open Source Bereich (MapServer, GRASS etc.). Leider ist entsprechendes Karten- und Luftbildmaterial außerhalb der USA nur sehr spärlich zu bekommen. Speziell in Deutschland hat sich noch nicht herumgesprochen, dass dieses Beharren auf kostenpflichtige Karten und Daten ein großes Innovationshemmnis ist⁶³.

59. cdli.mpiwg-berlin.mpg.de

60. worldwind.arc.nasa.gov

61. Allerdings haben die NASA und Google in 2006 eine weitgehende Kooperation vereinbart, so dass die Zukunft von *World Wind* etwas ungewiss ist.

62. Ein schon lange angekündigter Client für MacOS X lässt leider seit Jahren auf sich warten.

63. Forschungseinrichtungen in den USA kommen kostenlos an alle GIS-Daten, Luftbildaufnahmen und Karten, die mit öffentlichen Mitteln erstellt wurden und können daher schneller und präziser forschen. Deutsche Ämter hingegen, die die Hüter der Messtischblätter und Luftaufnahmen sind, verlangen teilweise Mondpreise für die Nutzung auch in der Forschung.

Für die Wissenschaft bedeutet dies daher, dass es – trotz meiner prinzipiellen Sympathie für Open Source – aus pragmatischen Gründen in vielen Fällen keine Alternativen zu *Google Earth* gibt⁶⁴. Jedoch sollte immer überlegt werden, ob nicht auf quelloffene Software ausgewichen werden und ob man eigenes Datenmaterial nicht der *OpenGIS*-Bewegung zur Verfügung stellen kann⁶⁵.

10. Datenschleuder 2.0

Die neuen Möglichkeiten des Web 2.0 bedingen aber auch neue Gefahren. Wer sich im Netz bewegt, hinterlässt deutlich sichtbar eine Datenspur. Diese Daten sind mehr oder weniger öffentlich und lassen sich mit *Data Mining*-Methoden leicht auslesen, rekombinieren und zu einem Gesamtbild zusammenfassen, das vielleicht mehr über einen verrät, als man gewillt ist preiszugeben. Hier hilft nur Aufklärung: Der *Net Citizen*, der Netzbürger, muss umfassend über die Gefahren aufgeklärt werden und wissen, wie er ein Ausprobieren seines Privatlebens verhindern kann.

Speziell die persönlichen Daten, die die Nutzer bei *Community Sites* wie *mySpace* oder *studiVZ* eingeben, sind geeignet, Begehrlichkeiten zu wecken. Da helfen auch *Datenschutzerklärungen* der Betreiber relativ wenig. Stellen wir uns mal vor, *studiVZ* wäre statt an Holtzbrinck (von denen ich hoffe und annehme, dass sie die deutschen Datenschutzbestimmungen noch ernst nehmen) zum Beispiel an einen amerikanischen Betreiber gegangen. Die Datenschutzerklärung wäre das (elektronische) Papier nicht wert gewesen, auf der sie steht.

Aber auch der Gesetzgeber ist gefordert: Gut gemeinte, aber schlecht realisierte Gesetze, wie zum Beispiel das Teledienstkommunikationsgesetz (TKG) oder der Mediendienststaatsvertrag (MDStV), die in ihrer aktuellen Fassung eine Impressumspflicht auch privater Webseiten (speziell Weblogs)

64. Um die Kosten zu minimieren, wäre es unter Umständen sinnvoll, wenn die MPG oder eines ihrer Rechenzentren **einen** *Google Earth*-Server für alle Institute aufstellen, betreuen und lizenzieren könnte.

65. Einen Überblick über Open Source GIS Tools bietet der hervorragende Band [[Mitchell 2005] Tyler Mitchell: *Web Mapping Illustrated. Using Open Source GIS Toolkits*, Sebastopol (O'Reilly) 2005], während [[Erle et al. 2005] Schuyler Erle, Rich Gibson & Jo Walsh: *Mapping Hacks, Tips & Tools for Electronic Cartography*, Sebastopol (O'Reilly) 2005] sowohl Open Source GIS als auch die Werkzeuge von Google und Yahoo! behandelt.

verlangen, stehen den berechtigten Schutzinteressen der meist privaten Weblogbetreiber gegenüber: Gerade weibliche und/oder minderjährige Weblogbetreiber sind mit der Preisgabe ihrer Adresse ein beliebtes Ziel für bestenfalls unseriöse Geschäftsmacher und schlimmstenfalls für Stalker. Und auch die immer wieder mal durch das Web schwappenden Abmahnwellen sind ein erstes und ernstzunehmendes Warnsignal.

10.1 Die Gefahr einer neuen dot.com-Blase

Wie wir gesehen haben, ist *Web 2.0* eigentlich nichts Neues. Es ist ein Begriff, unter dem die diversen Aspekte der *Social Software* ebenso subsummiert werden, wie das Zusammenführen von Techniken, die den Desktop im Webbrowser ersetzen sollen. *Web 2.0* ist daher eher ein klassisches *Buzz Word*. Und wie alle *Buzz Words* ruft es auch sofort die Spekulanten auf den Plan, die damit einen schnellen Euro machen wollen. Dies ist eine doppelte Gefahr: Zum einen kann das Interesse des großen Geldes, das mit aller Macht in den neuen Markt drücken will, die – wie wir (hoffentlich!) gezeigt haben – durchaus vorhandenen emanzipatorischen Möglichkeiten, die in den neuen Formen des Netzpublizierens stecken, an den Rand oder ganz ins Abseits drängen. Zum zweiten kann eine sicher bald einsetzende Enttäuschung darüber, dass man mit dem *Web 2.0* doch nicht so schnell wie Dagoberth Duck im Geld schwimmen kann, die positiven Möglichkeiten der *Social Software* überdecken.

Aber ich hoffe, dass das emanzipatorische Potenzial, das in den neuen Formen des Publizierens steckt, diese Angriffe überleben wird.

11. Anforderung an Werkzeuge für die Wissenschaft

Die unter dem *Buzz Word* *Web 2.0* subsummierten Techniken und Methoden bieten nicht nur – wie wir gesehen haben – die Möglichkeiten einer neuen Kommerz-Blase, sondern sie ermöglichen gleichzeitig (Dialektik, ick hör dir trappen) ein *grassroot web*, in der auch kleine Gruppen und Organisationen oder Einzelpersonen ohne großen Aufwand publizieren und kommunizieren können. Allerdings darf man dazu die weitere Entwicklung nicht dem „Markt“ alleine überlassen, sondern muss aktiv an der Gestaltung mitwirken. Ich möchte daher im Folgenden ein Modell für ein *grassroot network* wissenschaftlichen Arbeitens und Publizierens vorschlagen, das sehr stark von den Ideen der *Participatory Culture Foundation (PCF)*⁶⁶, den Betreibern der „freien“ Internet-TV-Plattform *Democracy* inspiriert wurde⁶⁷.

66. www.participatoryculture.org

Zusätzlich wurde dieses Modell von Jon Udell (vgl. [[Udell 1999] Jon Udell: Practical Internet Groupware. Building Tools for Collaboration, Sebastopol, CA (O'Reilly) 1999]) beeinflusst, der schon vor einigen Jahren zeigte, wie man mit vorgefundenen Werkzeugen, verbunden durch kleine *Glue-Scripte* eine *Groupware-Lösung* nicht nur alleine für die Redaktion des *BYTE Magazines*, sondern auch für die Kommunikation mit den Lesern schaffen konnte. Dieser vorhandenen Werkzeuge integrierende Ansatz soll hier fortgeführt werden.

11.1 Die Frage nach der Plattform

Sicher kann sich ein interessierter und halbwegs mit dem Internet vertrauter Wissenschaftler auch selber aus den bestehenden Web-2.0-Diensten (Flickr, YouTube, Sevenload etc.) eine persönliche Arbeitsumgebung „zusammenklicken“, so wie ich mir zum Beispiel bei Sevenload (Abb. 7) eine Materialsammlung eingerichtet habe, doch treten hierbei zwei Probleme auf:

Datenschutz: Es bestehen – meiner Meinung nach zu Recht – Bedenken bezüglich des Datenschutzes. Das betrifft sowohl die persönlichen Daten wie auch die hochgeladenen Exponate und Texte. Das ist noch wenig problematisch bei frei verfügbaren Daten, wird aber schnell ein Problem, wenn das hochgeladene Material copyrightbehaftet ist oder sonstigen Urheber- und Verwertungsrechten unterliegt. Und das dürfte – außer bei den historischen Wissenschaften – für die Mehrzahl der wissenschaftlichen Daten zutreffen.

Langzeitverfügbarkeit: Ebenso stellt sich die Frage, wie lange für einen Wissenschaftler die Daten gespeichert werden. Gerade kostenlose Dienste können von einem Tag auf den anderen gekündigt werden und auch kostenbehaftete Dienste sind zum Beispiel gegen einen Konkurs des Dienstleisters nicht gefeit.

67. In Grundzügen hatte ich dieses Modell aber auch schon auf meiner Keynote zur *BlogTalk 1* unter dem Title *Vom Weblog lernen... Community, Peer-to-Peer und Eigenständigkeit als ein Modell für zukünftige Wissenssammlungen* im Mai 2003 vorgestellt.

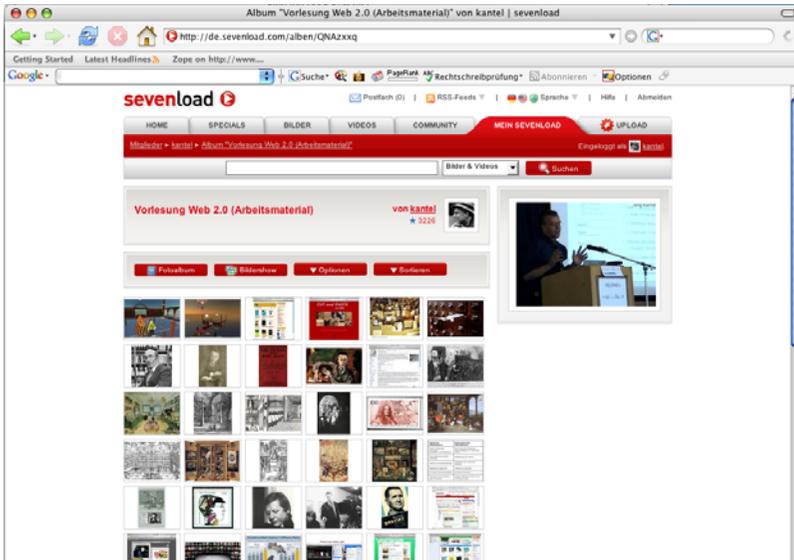


Abb. 7: In diesem *Album* bei Sevenload habe ich zum Beispiel alle meine Materialien (Screenshots, Photos, Zeichnungen, Videos etc.) für die Folien zu meiner Vorlesung „Web 2.0 – the two-way-web?“ gesammelt (de.sevenload.com/alben/QNAzxxq).

Diese zwei Probleme könnten gemindert werden, wenn die MPG (oder eines ihrer Rechenzentren, wie zum Beispiel die GWDG) selber als Diensteanbieter für die Wissenschaft aufträte. Ich denke, der einzelne Wissenschaftler hätte ihr gegenüber ein größeres Vertrauen als gegenüber kommerziellen Diensteanbietern.

Darüberhinaus müsste aber auch eine Langzeitverfügbarkeit der Daten gesichert sein. Das bedeutet, wenn die MPG als Diensteanbieter auftritt, dass sie diese Dienste den Wissenschaftlern auch weiterhin zur Verfügung stellt, wenn sie die Institute der MPG wieder verlassen haben. Schließlich sind viele Wissenschaftler nur für kurze Zeit an einem Institut der MPG und es besteht kein Anreiz, an einem solchen Experiment mitzuwirken, wenn die Daten nach dem Ausscheiden des Wissenschaftlers in den Archiven vermodern.

Daraus ergeben sich weitere Folgerungen: Um zum Beispiel eine fächerübergreifende Vernetzung zu erreichen, muss ein zentraler *Community Server* ähnlich Technorati existieren, der die Updates der angeschlossenen Personen und Gruppen sowie die Verlinkung und Rückverlinkung anzeigt,

Inhaltsverzeichnisse und Listen generiert und das Suchen in den Seiten der angeschlossenen Community-Mitglieder erlaubt.

Im übrigen Bereich sollte zweigleisig verfahren werden: Einmal ist es durchaus sinnvoll, Dienste, wie sie zum Beispiel YouTube, Flickr oder Sevenload anbieten, zentral anzubieten, andererseits sollte den beteiligten Wissenschaftlern, Gruppen und Instituten auch die Möglichkeit eingeräumt werden, eigene dezentrale Dienste hochzuziehen. Das entspräche der Software *Broadcast Machine* des *Democracy Projects*. Sie kann – im Prinzip – sowohl als zentraler Host für Tausende von Mitglieder eingesetzt, als auch als persönliches Repositorium eines Einzelnen genutzt werden.

Außerdem sollten alle Komponenten über ein abgestuftes System der Zugriffsrechte verfügen, das vom nur persönlichen Gebrauch über Nutzung von allen Gruppenmitgliedern, allen Institutsangehörigen, eine MPG-weite Freigabe bis hin zu einer Freigabe für die „ganze Welt“ reicht. Es versteht sich von selber, dass diese Freigaben für Lese- und Schreibrechte getrennt gesetzt werden können.

11.1.1 Weblog und Wiki

Das trifft besonders für Weblog- und Wikisoftware zu. Hier muss zudem auf eine große Flexibilität geachtet werden, damit der Nutzer durch die Software nicht eingeengt wird. Daher ist meiner Meinung nach einem kleinen (weblogfähigen) CMS der Vorzug vor einer spezialisierten Weblogsoftware zu geben. Kandidaten dafür wären zum Beispiel:

Drupal⁶⁸: Dieses CMS (Open Source, GPL) sticht durch besondere Community-Funktionen sowie einer nahezu unbegrenzten Erweiterbarkeit hervor. Es ist in PHP geschrieben und benötigt MySQL als darunterliegende Datenbank.

Wordpress⁶⁹: Dies ist eher ein CMS-fähiges Weblog-System (ebenfalls Open Source, GPL) denn ein vollwertiges CMS, besticht aber durch seine geringen Anforderungen (PHP, MySQL) an den darunterliegenden Server und ebenfalls durch seine große Flexibilität.

Beide Systeme sind durchaus in der Lage, auch als Hosting-System zu fungieren, doch gäbe ich für diesen Zweck der Twoday-Software⁷⁰ den Vorzug.

68. www.drupal.org

69. www.wordpress.org

Sie beruht auf dem in Java geschriebenen Web-Application-Server *Helma*⁷¹, ist ebenfalls unter der GPL als Open Source Software freigegeben und hat ihre Leistungsfähigkeit schon vielfach auch im kommerziellen Einsatz unter Beweis gestellt.

Für Wikis sollte ebenfalls das zweigleisige Verfahren eingehalten werden. Persönliche und Gruppen-Wikis sollten ebenso möglich sein wie zentrale, gehostete Lösungen. Dabei ist auf jeden Fall auf die *Interwiki*-Fähigkeit zu achten, die die relativ problemlose Vernetzung einzelner Wikis untereinander ermöglicht. Von dem Einsatz des *MediaWikis*⁷² würde ich wegen der hohen Komplexität und der etwas schwierigen Installation und den hohen Systemanforderungen eher abraten⁷³ und auf eine kleinere Lösung setzen. Hier wäre besonders zu diskutieren, wieweit man auf das *TWiki*⁷⁴-Know-How des MPI für Kognitions- und Neurowissenschaften in Leipzig zurückgreifen kann.

11.1.2 Ein kollektives Repositorium

Zentraler Punkt meiner Vorschläge ist aber die Schaffung einer Möglichkeit für Wissenschaftler, Sammlungen anzulegen und zwar sowohl zentral, als auch in Gruppen oder als Einzelner. Hier kann man bisher aber leider nicht auf fertige Software zurückgreifen, sondern bestenfalls sich Ideen von bisherigen Lösungen ausborgen⁷⁵:

Was sollte gesammelt werden können? Aus meinen eigenen Erfahrungen heraus halte ich es nicht für sinnvoll, spezialisierte Lösungen nur für Photos,

70. www.twoday.net

71. www.helma.at

72. www.mediawiki.org

73. Aber hier gilt natürlich das Gleiche, wie für die gesamten Vorschläge in diesem Abschnitt: Sie bedürfen noch der Diskussion.

74. www.twiki.org

75. Wobei die Lösung, die Sevenload anbietet, meinen Ideen wegen der Möglichkeit, sowohl Bilder als auch Videos dort zu sammeln, noch am Nächsten kommt. Allerdings ist es einmal eine rein zentrale Lösung, verteiltes Sammeln ist dort nicht möglich. Und es fehlt mir die Möglichkeit, auch Texte dort einzustellen. Einen weiteren Teil der Ideen habe ich dem *BSCW-System (Basic Support for Collaborative Work)* der Fraunhofer Gesellschaft entnommen.

nur für Videos oder nur für Texte anzulegen. Gesammelt können sollte alles, was sich in irgendeiner Form digitalisieren und auf einem Rechner ablegen läßt. Messdaten zum Beispiel sollten **neben** den Visualisierungen der Messdaten (seien es Bilder oder Filme) und ihrer textuellen Auswertung und der Veröffentlichung darüber liegen und nicht jeweils getrennt auf gesonderten Servern⁷⁶.

Last but not least sollten bestehende und bewährte Applikationen integriert werden. So etwas wie die *digilib* (vgl. [[Graßhoff et al. 2001] Gerd Graßhoff, Hans-Christoph Liess und Kärin Nickelsen: COMPAGO. Der systematische Bildvergleich, Bern (Bern Studies in History of Philosophy of Science) 2001]), die im Rahmen der ECHO-Initiative entwickelt wurde und (nicht nur) dort eingesetzt wird, kann und sollte durchaus als Basistechnologie eingebunden werden.

Was schwebt mir also vor? Eine Software ähnlich Flickr, YouTube oder Sevenload, die einmal alle Arten von Datenmaterial speichern, sammeln und mit Tags versehen kann, die aber sowohl als Host zentral betrieben als auch als selbständige Applikation auf einem eigenen Server lauffähig ist. Diese einzelnen Server sollten sowohl die nötigen „klassischen“ *Community-Funktionen* (Kommentare, Ping, Trackback, RSS) bieten als auch untereinander vernetzt sein, so dass zum Beispiel eine Suche über alle angeschlossenen Server möglich ist. Den Grad der Vernetzung sollte der jeweilige Wissenschaftler oder die jeweilige Projektgruppe selber bestimmen können.

Und sie sollte natürlich Schnittstellen (APIs) zur Verfügung stellen, die es den Wissenschaftlern erlauben, via Script- und/oder Programmiersprachen

76. Mir ist natürlich klar, dass dies gerade bei Messdaten schnell die Kapazität einzelner Server sprengen könnte, dann sollte aber wenigstens der Link zu diesen Daten im Repositoryum liegen.

auf die Funktionen dieser Server zuzugreifen und sie in eigene (Web-) Applikationen einzubinden.



Abb. 8: Frans Francken d. J. (1581 - 1642): Kuriositätenkabinett, nach 1636, nach anderen Quellen um 1620 - 25, Kunsthistorisches Museum Wien

So haben wir im schlimmsten Falle weiterhin die private Sammlung, die aber vom jeweiligen Sammler – sollte er dann doch einmal von den Vorzügen der Kollaboration überzeugt sein – mit wenigen Klicks in die gemeinschaftliche Sammlung überführt werden kann. Und vielleicht entsteht dadurch im Laufe der Zeit so etwas wie eine vielfältig vernetzte und verlinkte, gemeinsam gepflegte Kunst- und Wunderkammer (vgl. zum Beispiel [[Bredekamp 2003] Horst Bredekamp: Gottfried Wilhelm Leibniz' Bildtheater des Wissens, in: Helmar Schramm et al. (Hrsg.): Bühnen des Wissens. Interferenzen zwischen Wissenschaft und Kunst, Berlin (dahlem university press) 2003]), eine verteilte Wissenssammlung, die durch Neuordnung

und Rekombination ihrer Exponate zu neuen Anregungen, Ideen und Erkenntnissen führt.

Literatur

[Alby 2007]Tom Alby: *Web 2.0. Konzepte, Anwendungen, Technologien*, München (Hanser) 2007

[Bredenkamp 2003]Horst Bredenkamp: *Gottfried Wilhelm Leibniz' Bildtheater des Wissens*, in: Helmar Schramm et al. (Hrsg.): *Bühnen des Wissens. Interferenzen zwischen Wissenschaft und Kunst*, Berlin (dahlem university press) 2003

[Bulterman, Rutledge 2004]Dick C. Bulterman, Lloyd Rutledge: *SMIL 2.0. Interactive Multimedia for Web and Mobile Devices*, Berlin und Heidelberg (Springer) 2004

[Diemand et al. 2007]Vanessa Diemand, Michael Mangold, Peter Weibel (Hrsg.): *Weblogs, Podcasting und Videojournalismus. Neue Medien zwischen demokratischen und ökonomischen Potenzialen*, Hannover (Heise) 2007

[ECHO 2001]Jürgen Renn (ed.) *ECHO – an infrastructure to bring European Cultural Heritage Online*, Berlin (MPIWG Preprint 191) 2000

[ECHO 2002]Max Planck Society and the Echo Initiative: *Towards a Web of Culture and Science* Berlin und München o.J. (2002)

[Eisenberg 2002]J. David Eisenberg: *SVG Essentials. Producing Scalable Vector Graphics with XML*, Sebastopol, CA (O'Reilly) 2002

[Englander 2002]Robert Englander: *Java and SOAP*, Sebastopol, CA (O'Reilly) 2002

[Erle et al. 2005]Schuyler Erle, Rich Gibson & Jo Walsh: *Mapping Hacks, Tips & Tools for Electronic Cartography*, Seabstopol (O'Reilly) 2005

[Franzmann 2006]Edgar Franzmann: *Weblogs, Podcasts & Co. Ein praktischer Leitfaden für den Umgang mit neuen Kommunikationswegen im Netz*, Ratgeber Neue Medien, Band 5, Köln (Landesanstalt für Medien NRW) 2006

[Geroimenko, Chen 2005]Vladimir Geroimenko, Chaomei Chen (Eds.): *Visualizing Information Using SVG and X3D. XML-based Technologies for the XML-based Web*, London, Berlin und Heidelberg (Springer) 2005

- [Gloor 1997]Peter Gloor: *Elements of Hypermedia Design. Techniques for Navigation & Visualization in Cyberspace*, Boston, MA (Birkhäuser) 1997
- [Graßhoff et al. 2001]Gerd Graßhoff, Hans-Christoph Liess und Kärin Nikelsen: *COMPAGO. Der systematische Bildvergleich*, Bern (Bern Studies in History of Philosophy of Science) 2001
- [Kantel 2006]Jörg Kantel: *Ping, Tags und Social Software – Communitybildung und Medienkonvergenz durch neue Publikationsformen im Internet*, in: Prof. Dr.-Ing. Klaus Regensburg (Hrsg.): *Workshop „Informationsgesellschaft“ der Stiftungskollegiaten des SVK Berlin*, Berlin 2006
- [Kantel 2007]Jörg Kantel: *RSS & Atom kurz und gut*, Köln (O'Reilly) 2007
- [Lange 2006]Corina Lange: *Web 2.0 zum Mitmachen*, Köln (O'Reilly) 2006
- [Mitchell 2005]Tyler Mitchell: *Web Mapping Illustrated. Using Open Source GIS Toolkits*, Sebastopol (O'Reilly) 2005
- [Pantle 2006]Henrik Pantle: *Beyond Broadcast - Rundfunk auf Abruf. Konvergenz von „Broadcast“ und „On Demand“ und die Chancen von User Generated Content*, in: Prof. Dr.-Ing Klaus Regensburg (Hrsg.): *2. Tagung Neue Medien und Technologien der Informationsgesellschaft 2006 „Film, Computer und Fernsehen“*, Berlin 2006
- [Plag, Riempp 2007]Florian Plag, Roland Riempp: *Interaktives Video im Internet mit Flash. Konzeption und Produktion von Videos für das WWW*, Berlin und Heidelberg (Springer) 2007
- [Renn 2000]Jürgen Renn: *Challenges of the Information Revolution for the Max Planck Society*, Berlin (MPIWG Preprint 151) 1999
- [Renn 2001]Jürgen Renn: *Erwirb es um es zu besitzen: Kulturelles Erbe im Zeitalter der Informationsrevolution*, Berlin (MPIWG Preprint 176) 2000
- [Salathé 2001]Marcel Salathé: *SVG für Windows und Macintosh*, München (Markt + Technik) 2001
- [Schröder 2007]Sebastian Schröder: *Web 2.0-Techniken für die CDLI (Cuneiform Digital Library Initiative)*, in diesem Band
- [Snell et al. 2002]James Snell, Doug Tidwell, and Pavel Kulchenko: *Programming Web Services with SOAP*, Sebastopol, CA (O'Reilly) 2002
- [Udell 1999]Jon Udell: *Practical Internet Groupware. Building Tools for Collaboration*, Sebastopol, CA (O'Reilly) 1999

Langzeitarchivierung digitaler Datenbestände an der Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen (GWDG) am Beispiel des Kunsthistorischen Instituts in Florenz – Max-Planck-Institut (KHI)

Timo Scheller

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen

Zusammenfassung

Der vorliegende Artikel stellt die Aktivitäten und den aktuellen Arbeitsstand im Bereich Langzeitarchivierung (LZA) an der GWDG vor. Gemeinsam mit dem Rechenzentrum Garching (RZG) wurde 2005 ein Antrag beim Beratern Ausschuss für EDV-Anlagen in der Max-Planck-Gesellschaft (BAR) gestellt, der eine Erweiterung der Speicherinfrastruktur an der GWDG zum Gegenstand hatte. Zu den wesentlichen Komponenten gehört ein zusätzlicher TSM-Server, zwei Archiv-Fileserver sowie Erweiterungen für das GWDG-SAN. Für das BMBF-Projekt kopal¹ betreibt die GWDG das Archivsystem und das entsprechende Speicher-Backend. Dieses Archivsystem

1. <http://kopal.langzeitarchivierung.de>

stem ist in die aktuelle Infrastruktur integriert und soll später an den beantragten LZA-eigenen Speicherbereich angebunden werden. Es wird zwischen der reinen Speicherung von Daten zum Erhalt des Datenstroms (Bitstream Preservation) und der Langzeitverfügbarkeit der Inhalte unterschieden. Lösungen für beide Ansätze werden vorgestellt. Am Beispiel des KHI wird der aktuelle Stand der Arbeiten für die Max-Planck-Gesellschaft (MPG) im Bereich Bitstream Preservation vorgestellt. Die GWDG archiviert bereits für einige Max-Planck-Institute (MPIe), z. B. das KHI, digitale Daten mit langfristiger Zielsetzung. Dies erfolgt aktuell noch dezentral und soll künftig über die beantragte neue LZA-Speicherinfrastruktur vereinheitlicht werden. Abschließend werden noch offene Herausforderungen an die GWDG im Bereich LZA aufgezeigt. Hierzu gehören sowohl technische als auch organisatorische Fragen. Die technischen Fragen beziehen sich vorwiegend auf Datentransfer und Datenspeicherung. Die organisatorischen betreffen unter anderem Kostenfragen und Servicemodelle.

1. Einleitung

Die GWDG befasst sich seit 2001 mit Fragen der LZA digitaler Datenbestände. Als ein Rechen- und Kompetenzzentrum für die MPG und Hochschulrechenzentrum für die Universität Göttingen sieht es die GWDG als ihre Aufgabe, ihren Kunden einen solchen Service anzubieten. Zu diesem Zweck wurde eine Reihe von Aktivitäten eingeleitet. Der aktuelle Arbeitsstand soll am Beispiel der LZA für das KHI in Florenz in diesem Beitrag vorgestellt werden.

2. Gemeinsamer BAR-Antrag von GWDG und RZG

Im Jahr 2005 haben die GWDG und das RZG erfolgreich einen gemeinsamen BAR-Antrag gestellt. Über diesen Antrag konnten die nötigen Hardware-Ressourcen für die Erfüllung des LZA-Auftrags durch die MPG bereitgestellt werden. Für die MPIe wurden insgesamt 50 TeraByte (TB) Speicherplatz beantragt, ca. 30 TB für 2006 und 20 TB für 2007. Dieser Mengenschätzung liegt ein Speicherkonzept zugrunde, das eine redundante Datenhaltung mit mindestens zwei Kopien in den Bandrobotern der GWDG und möglichst eine zusätzliche Kopie am RZG vorsieht. Gegenstand der Beantragung war die erforderliche Hardware zum Aufbau eines speziellen LZA-Speicherbereichs an der GWDG. Dieser wird eine einheitliche LZA-Lösung im Speicher-Backend für die Daten der MPG, der Universität Göttingen und des kopal-Projekts darstellen. Ein solches gemeinsames Speicher-Backend ermöglicht maximale Synergien bei der Archivierung aller Langzeitbestände.

2.1 Basisinfrastruktur GWDG

Die nachstehende Skizze zeigt die GWDG-Speicherinfrastruktur vor der Erweiterung durch die beantragten Hardwarekomponenten.

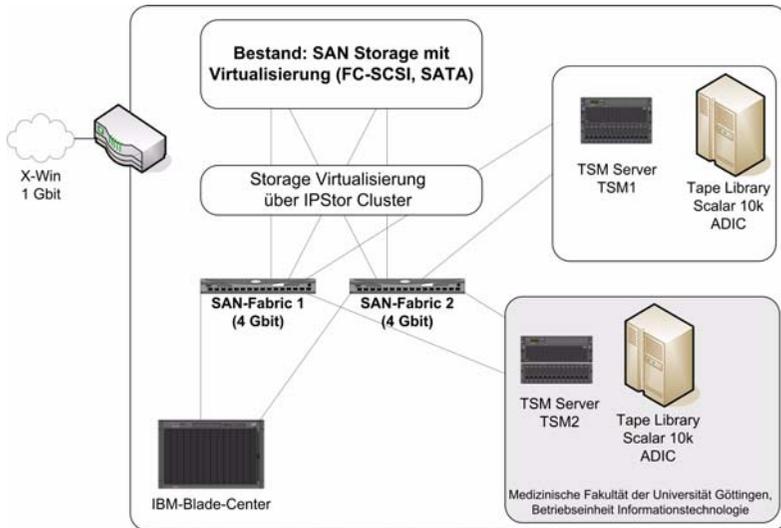


Abb. 1

Die ursprüngliche GWDG-Speicherinfrastruktur umfasste zwei TSM-Server und zwei ADIC Scalar 10k Tape Libraries an getrennten Standorten in Göttingen, der GWDG selbst und dem Göttinger Klinikum, beide ca. drei Kilometer Luftlinie voneinander entfernt. Die TSM-Server als auch die Bandbibliotheken sind an das SAN der GWDG mit unterschiedlichsten Plattenspeichern über eine vollständig redundante Fabric angebunden. Das GWDG-SAN ist mit der Speichervirtualisierungssoftware von IPStor ausgestattet. Bereits vorhanden war ebenfalls ein IBM-Blade-Center, das noch über freie Einschübe verfügte, die für die LZA-Erweiterung genutzt wurden.

2.2 Infrastruktur GWDG mit Erweiterung BAR-Antrag

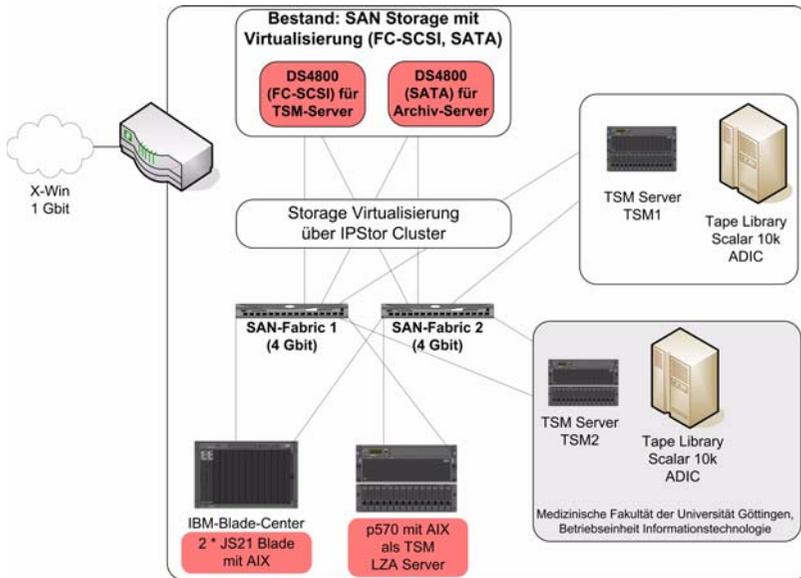


Abb. 2

Zu den über den BAR-Antrag erworbenen Komponenten gehört ein zusätzlicher TSM-Server, ein IBM p570, der analog zu den bereits vorhandenen TSM-Servern der GWDG unter AIX betrieben und dessen Konfiguration an den Anforderungen der LZA ausgerichtet werden wird. Für diesen LZA-TSM-Server wurde ein DS4800 FC SCSI Disk System beantragt, das eine Maximalkapazität von 10 TB bietet. Es ist vorwiegend für die TSM-Datenbank und für Caching-Funktionen eingeplant. Für die Verwaltung der LZA-Daten wurden zwei JS21-Blade-Server beantragt. Die Aufgabe dieser Archiv-Fileserver wird den Datentransfer mit den Instituten, den Zugriff auf die Daten und ggf. Zusatzfunktionen zur Sicherung von Datenintegrität und -authentizität, wie Checksummen oder Virenkontrollen, umfassen. Zum Ausbau des SANs selbst wurden vier 4-Gbit-FC-Switche eingeplant und 24 TB Speicherkapazität auf SATA-Platten.

2.3 Infrastruktur GWDG mit Erweiterung BAR-Antrag und kopal-Integration

Die GWDG beteiligt sich am BMBF-geförderten Projekt kopal – „Kooperativer Aufbau eines Langzeitarchivs digitaler Informationen“. Ziel dieses im Juli 2004 begonnenen dreijährigen Projekts ist der Aufbau einer technischen

und organisatorischen Lösung, um die Langzeitverfügbarkeit elektronischer Publikationen zu sichern. Der Systembetrieb erfolgt an der GWDG. Das kopal-System ist direkt in die vorhandene Speicherinfrastruktur der GWDG eingebunden und soll künftig die neuen LZA-spezifischen Komponenten, insbesondere den LZA-eigenen TSM-Server nutzen. Die untenstehende Skizze zeigt die aktuelle Einbindung des kopal-Systems bestehend aus fünf IBM p550 Servern, zwei für den Test- und drei für den Produktivbetrieb. Derzeit erfolgt die Speicherverwaltung noch über den am Göttinger Klinikum befindlichen TSM-Server in Form einer eigenen Serverinstanz.

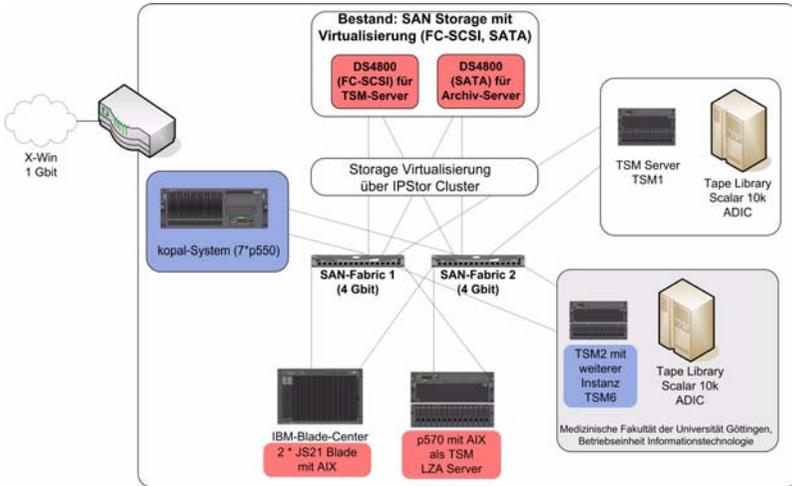


Abb. 3

3. Bitstream Preservation vs. Langzeitverfügbarkeit Zwei unterschiedliche Aufgaben der LZA

Bei der langfristigen Sicherung von Datenbeständen sind zwei Aspekte zu berücksichtigen. Zum einen müssen die Daten langfristig sicher gespeichert werden, d. h. die „Bits n Bytes“ müssen unbeschädigt erhalten werden. Hierfür ist eine vorausschauende Planung und Verwaltung der eingesetzten Speichertechnik nötig. Hierzu gehört eine Auseinandersetzung mit Fragen nach Lebensdauer und Zuverlässigkeit sowie, wenn nötig, Erneuerbarkeit oder Austauschbarkeit eingesetzter Speichertechnik- und Medien. Neben diesem physikalischen Erhalt von Daten ist für eine langfristige Verfügbarkeit der Inhalte auch ihre Interpretierbarkeit durch entsprechende Darstellungsprogramme erforderlich. Diesem Zweck dienen Formatmigrationen, die die Aktualität des Dateiformats und damit seine Lesbarkeit sicherstellen. Auch

Emulationen von Interpretationsumgebungen können hier als Option genannt werden. Der zuverlässige Erhalt des physikalischen Datenstroms (Bitstream Preservation) ist immer die Voraussetzung aller weiteren Aktivitäten und daher von zentraler Bedeutung für die LZA digitaler Daten.

3.1 Bitstream Preservation an der GWDG

Ein Service für die MPG und die Universität Göttingen

Die GWDG bietet für die Institute der MPG und der Universität Göttingen eine langfristig sichere Speicherung im Sinne der oben beschriebenen Bitstream Preservation an. Dieser Service nutzt die durch den BAR-Antrag beschafften Hardwarekomponenten. Es handelt sich um einen Basisdienst, der die Übernahme und je nach Anforderungen mehrfach redundante Speicherung der Daten in einem Verzeichnissystem auf den JS21-Archivfileservern umfasst. Eine spezielle Software ist weder auf Seiten der GWDG noch auf der der Institute erforderlich. Der Datentransfer kann über verschiedene standardisierte Protokolle wie scp, sftp, oder rsync erfolgen. Im Bedarfsfall wird eine Integritätskontrolle der Daten über Checksummen vorgenommen. Weitere Zusatzfunktionen, wie z. B. Virenkontrollen, sind geplant. Auf dem Archiv-Fileserver selbst ist ein TSM-Client installiert, der die Daten an den TSM-Server zur langfristigen Sicherung weitergibt. Diese Anbindung soll später um eine HSM-Funktion für bestimmte Verzeichnisse erweitert werden.

3.2 Langzeitverfügbarkeit digitaler Daten an der GWDG

Das Projekt kopal an der GWDG

Zusätzlich zu den Aktivitäten der GWDG im Bereich Bitstream Preservation betreibt die GWDG das Langzeitarchivsystem des BMBF-Projekts kopal². Dieses Archivsystem bietet über die reine Sicherung des Bitstreams hinaus Techniken und Verfahren, die auch die langfristige Interpretierbarkeit der Archivbestände gewährleisten. Hierzu gehören z. B. das Erstellen und Speichern technischer Metadaten als Informationsgrundlage für spätere Maßnahmen zum Erhalt der Lesbarkeit der Daten. Solche Maßnahmen können Formatmigrationen oder die Bereitstellung von Emulationsumgebungen sein. An der Anbindung von Migrationstools oder Emulationsunterstützungen wird im Rahmen des Projekts gearbeitet. Das Archivsystem besteht aus einer Client-Server-Umgebung. Auf Server-Seite,



2. <http://kopal.langzeitarchivierung.de>

GWDG, wird das IBM-System DIAS betrieben, das auf den Standardsoftwarekomponenten DB2, Content Manager und Websphere Application Server basiert. Auf Client-Seite, Deutsche Nationalbibliothek (DNB) und Niedersächsische Staats- und Universitätsbibliothek (SUB), wird die koLibRI-Software eingesetzt. Hierbei handelt es sich um verschiedene Open Source Java-Bibliotheken, die im Rahmen des kopal-Projekts von den Projektpartnern DNB und SUB entwickelt wurden.

Die koLibRI-Software übernimmt in erster Linie das Erstellen, Einbringen und Abrufen von kopal-konformen Archivpaketen in das DIAS-System an der GWDG. Beim Erstellen der Pakete wird das im kopal-Projekt entwickelte Universal Object Format (UOF) verwendet, das insbesondere solche Metadaten berücksichtigt, die der Langzeitverfügbarkeit der Daten dienen. Die koLibRI-Software dient aber auch der Integration des kopal-Systems in bestehende Workflows der nutzenden Bibliotheken.



Die Anbindung an die Speicherinfrastruktur der GWDG erfolgt über die TSM-Client-Software. Bei der Datensicherung werden drei Bereiche, Systembackup, Datenbank-Backup und offline-Speicherung der Archivpakete selbst, berücksichtigt. Das System- und Datenbank-Backup wird über einen TSM-Backup-Client auf den kopal-Servern ausgeführt. Die Archivpakete selbst werden über die TSM-Schnittstelle des Content Manager (TSM-Client API) an den TSM-Server weitergereicht und von diesem in vierfacher offline-Haltung in den Bandrobotern verwaltet.

4. Aktuelle Verfahren des Datentransfers und der Datenhaltung für das KHI (Stand November 2006)

Der Datentransfer erfolgt aktuell über den Postversand von USB-Platten. Dies ist zurzeit der einfachste Weg, da zum einen die Anbindung des KHI an das Internet den zum Teil sehr schwankenden Performanceanforderungen des Datentransfers nicht entspricht. Zum anderen wird dieses Verfahren noch genutzt, weil das geplante rsync-Verfahren einen einheitlichen, auf beiden Seiten gleichen Verzeichnisbaum voraussetzt, der erst mit dem Wechsel auf die neue Speicherinfrastruktur realisiert werden wird. Tests des Datentransfers mittels rsync-Verfahren über ISDN-Leitung verliefen sehr positiv und eine Umstellung ist beabsichtigt, sobald der einheitliche Verzeichnisbaum im neuen LZA-Speicherbereich der GWDG erstellt und mit dem des KHI in Florenz abgeglichen werden konnte. Das derzeitige Datenvolumen des KHI an der GWDG beträgt ca. 4,5 TB. Die Datenbestände wachsen stetig und

ändern sich teilweise noch nach der Übertragung an die GWDG. Dieser Umstand erfordert eine sehr flexible Lösung für das KHI, an der aktuell gearbeitet wird. Derzeit befinden sich die Daten auf verschiedenen Plattenbereichen. Sie sind aufgeteilt in viele Unterverzeichnisse, die den Einteilungen auf den überstellten USB-Platten entsprechen. Die Daten werden mittels Checksummen beim Kopieren mittels Skript Routinen auf ihre Integrität überprüft. Die Ergebnisse werden in Log-Files protokolliert.

5. Künftige Verfahren des Datentransfers und der Datenhaltung für das KHI

Auf der neuen LZA-Speicherinfrastruktur der GWDG werden alle Daten in einem einheitlichen Plattenbereich entsprechender Größe verwaltet werden. Hierfür ist eine Konsolidierung der Daten erforderlich. Hierzu ist ein Umkopierprozess aller vorhandenen Daten von ihren jetzigen Speicherorten in den neuen LZA-Speicherbereich erforderlich, der bereits begonnen wurde. Sobald dieser abgeschlossen ist, ist eine Umstellung der Transfermethode auf einen regelmäßigen rsync-Prozess geplant. Um dem Umstand Rechnung zu tragen, dass sich Daten auch nach der Überstellung an die GWDG teilweise noch ändern, wird derzeit eine HSM-Lösung evaluiert. Ein Teil der Daten des KHI soll, außer für die LZA, auch für den Webzugriff der Digitalen Photothek³ verfügbar sein. Zu diesem Zweck wird von diesem Teilbestand zusätzlich eine Zugriffskopie angelegt. Wenn die Übertragung der Bestände abgeschlossen ist, wird die endgültige Archivierung im neuen Speicherbereich eingeleitet werden. Die Daten werden dann in mehrfacher Redundanz auf beiden Bandrobotern der GWDG an getrennten Lokalitäten gesichert werden. Eine Spiegelung dieser Bestände an einem dritten Standort, dem Rechenzentrum Garching (RZG), ist zusätzlich in Planung.

6. Aktuelle Herausforderungen im Bereich LZA an der GWDG

Eine zentrale Fragestellung für die langfristige Archivierung von Daten betrifft ihre spätere Änderbarkeit. Hier muss für das jeweilige Institut eine Lösung gefunden werden, wann welche Bestände sinnvoll in eine dauerhafte Archivierung überstellt werden sollen und wie ggf. Zugriffsrechte, insbesondere Schreibrechte, zulässig sein sollen. Eine weitere Herausforderung besteht in der Handhabung sehr heterogener Datenbestände. Die zu archivierenden Bestände setzen sich zum Teil aus sehr unterschiedlich großen

3. Digitale Photothek des KHI: <http://www.khi.fotothek.org/>

Dateien zusammen, die sich in einer ggf. zu erhaltenden Verzeichnisstruktur befinden. Zurzeit bietet die GWDG noch unterschiedliche Datentransferverfahren an. Hier ist in den kommenden Monaten zu prüfen, ob ggf. auf ein ausgewähltes Verfahren reduziert werden kann, das besonderen Sicherheitsstandards entspricht und Kontrollen für Datenintegrität und -authentizität wie Checksummen und Virenkontrollen einbindet. Auch werden derzeit verschiedene Verfahren evaluiert, die Daten an den Bandroboter zur offline-Sicherung zu übergeben. Hierzu zählen klassische Archivierungs- und Backupverfahren sowie die Nutzung von HSM-Lösungen. Verschiedene Redundanzszenarien und der Einsatz zusätzlicher Speichermedien werden in diesem Zusammenhang ebenfalls geprüft. An diese Überlegungen schließt sich auch die Erarbeitung und Implementierung von gezielten Verfahren zu Aktualisierung verwendeter Speichertechniken und -medien an. Ziel ist es eine zuverlässige Medien-Migrations- bzw. Refreshment-Policy an der GWDG zu entwickeln. Die Einhaltung einer solchen Policy stellt eine zentrale Anforderung an die nachweisbare Vertrauenswürdigkeit eines digitalen Langzeitarchivs dar. Solchen Anforderungen, wie sie z. B. seitens nestor, dem Kompetenznetzwerk Langzeitarchivierung, und anderen Institutionen aktuell formuliert werden, zu genügen, ist eine der organisatorischen Herausforderungen im Bereich LZA. Weitere organisatorische Herausforderungen sind die Formulierung unterschiedlicher Serviceangebote, aus denen je nach Kundenanforderung eine Auswahl getroffen werden kann, und die Kostenberechnung und -kontrolle für den zunehmenden technischen und administrativen Aufwand.

7. Ausblick

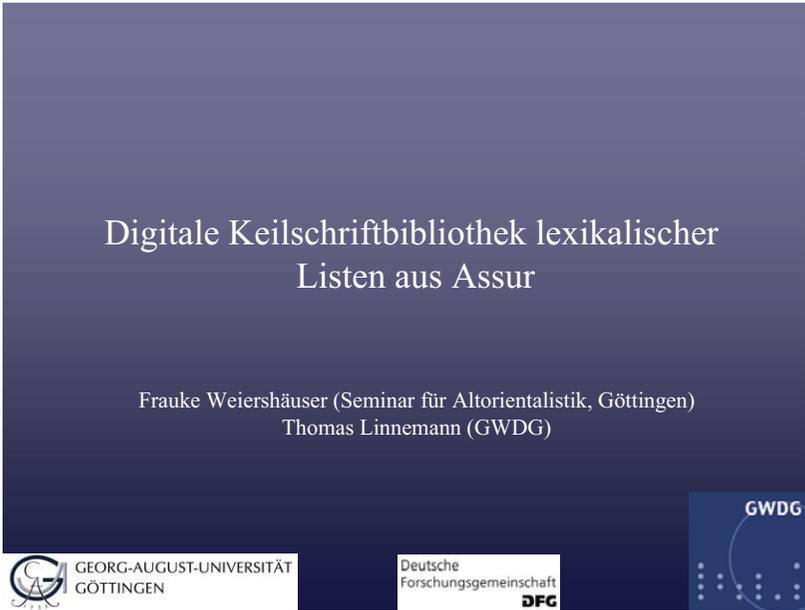
Die GWDG wird an allen genannten Punkten kontinuierlich weiterarbeiten. Der Schwerpunkt der Aktivitäten wird im Ausbau eines stabilen, flexiblen und skalierbaren Speicherbereichs für die LZA liegen. Darüber hinaus werden der Weiterbetrieb des kopal-Systems und seine Nutzung durch andere Institute, insbesondere der MPG und der Universität Göttingen, angestrebt. Die GWDG wird sich aktiv an allgemeinen Forschungsarbeiten, Konferenzen und Workshops zum Thema LZA beteiligen. Für den Standort Göttingen beteiligt sich die GWDG an einer LZA-Arbeitsgruppe im Rahmen des GÖ*-Projekts.

Digitale Keilschriftbibliothek lexikalischer Listen aus Assur

Frauke Weiershäuser, Thomas Linnemann

Seminar für Altorientalistik der Universität Göttingen

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen



Digitale Keilschriftbibliothek lexikalischer Listen aus Assur

Frauke Weiershäuser (Seminar für Altorientalistik, Göttingen)
Thomas Linnemann (GWDG)

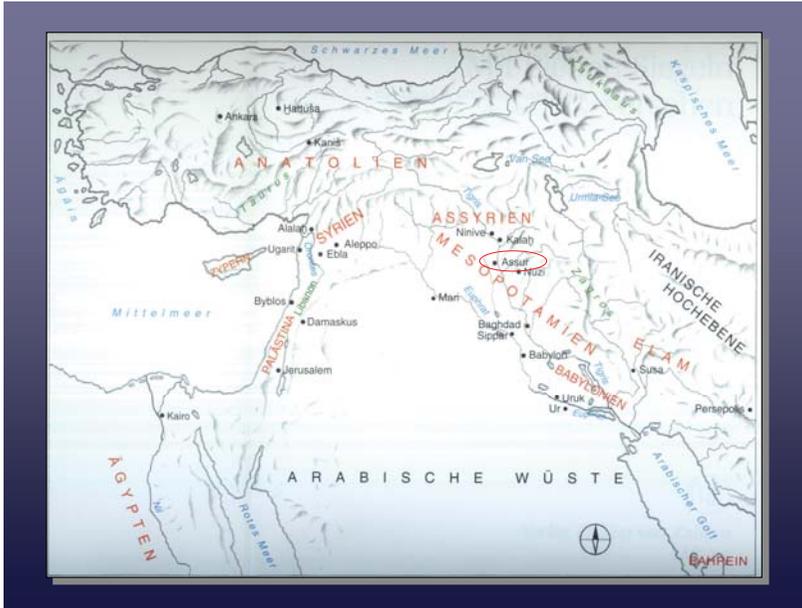


GEORG-AUGUST-UNIVERSITÄT
GÖTTINGEN

Deutsche
Forschungsgemeinschaft
DFG



Ein Projekt der Universität Göttingen und der GWDG, gefördert durch die die DFG.



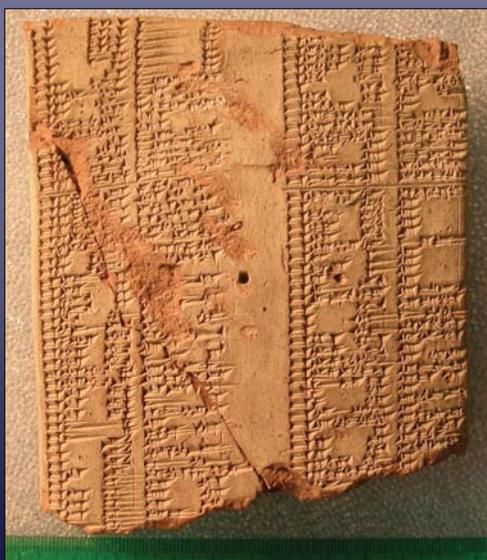
Seit 1997 werden die bisher noch unpublizierten Funde aus den 1903 - 1914 durchgeführten Ausgrabungen in der antiken assyrischen Königsstadt Assur im Rahmen eines großangelegten DFG-Projektes in diversen Teilprojekten bearbeitet und veröffentlicht.



Rekonstruktionszeichnungen
der Stadt Assur
von Walter Andrae



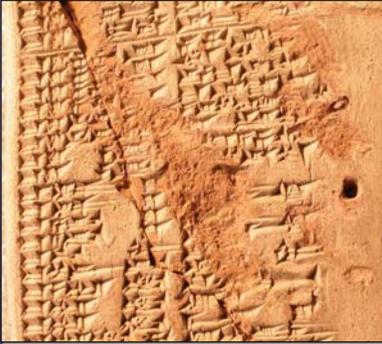
Die im Rahmen des Göttinger Projektes bearbeiteten Texte stammen aus der Zeit von ca. 1400 - 650 v. Chr. Die Stadt Assur, hier Rekonstruktionszeichnungen des Ausgräbers, war bis zu Beginn des 9. Jahrhunderts v. Chr. Hauptstadt des assyrischen Reiches, danach jedoch stets noch ein wichtiges kultisches Zentrum des Landes.



VAT 9713
2. Hälfte 2. Jt. v. Chr.

Es werden Waffen
und Werkzeuge
aus Holz aufgelistet.

Bei den Texten aus Assur handelt es sich um Keilschrifttexte, die in den noch feuchten Ton gedrückt wurden. In Göttingen wird die Gruppe der so genannten „Lexikalischen Listen“ bearbeitet. Dabei handelt es sich um zumeist zweisprachige Listen einzelner Worte oder auch komplexerer Ausdrücke, also gleichsam um antike mesopotamische Wörterbücher.



Ausschnitt aus VAT 9713
Foto und Handkopie

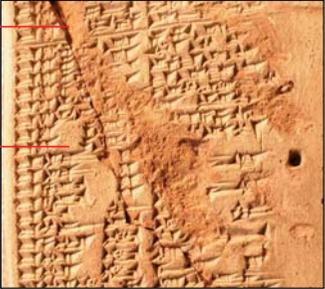
In der traditionellen Form der Publikation werden von den Keilschrifttafeln Handkopien angefertigt, bei denen die dreidimensionalen Eindrücke der Keilschriftzeichen abgezeichnet werden.

19 ^d	ḡ ⁸⁸ tukul-[sag]-zag-na ₄	ḥu-tap- <i>pa-lu-ú</i> ⁷	AHW 356 ḥuppālā: Doppelaxt CAD H 263 ḥutpalā: mace ⁵

20	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa ¹ -ṭu	za-ḥa-ṭu- <i>ú</i> ⁷	AHW 1503 zaḥaṭū: eine Streitaxt CAD Z 13 zaḥaṭū: battle-axe
21	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa ¹ -ṭu	qūl-mu- <i>ú</i> ⁷	AHW 927 qulmū: eine Axt CAD Q 299 qulmū: (an ax)
22	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa-ṭu ¹	kal-ma-ak- <i>ru</i> ⁷	AHW 426 kalmak/qrū: eine Streitaxt CAD K 86 kalmakru: battle-ax
23	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa-ṭu ¹	ma-ag-ša- <i>ru</i> ⁷	AHW 577 maḡšaru: sub 2) eine Streitaxt CAD M/1 48 maḡšaru: (an ax)
24	ḡ ⁸⁸ tukul 'kun gar-ra ¹	šá zi-ba-ta [ár]- <i>ru</i> ⁷	AHW 69 armu 1: bedeckt AHW 1524 zibbatu: Schwanz, sub B 2) unkl. in Bez. v Schwert CAD A/2 228 arāmu: mace covered with a "tail" CAD Z 100 zibbatu: tail
25	ḡ ⁸⁸ tukul 'kun si ¹ -ga	'KI-MIN ¹ saḥ- <i>ru</i> ⁷	AHW 1004 saḥāpu: sub 9) überdecken (mit) b) (Waffe) CAD S 31 saḥāpu: to cover "Waffe, die mit einem 'Schwanz' bedeckt ist"
26	ḡ ⁸⁸ tukul niḡ- <i>ʿ</i> sik ¹	[kak]-ki a-za- <i>ru</i> ⁷	AHW 1511 zappu: B ein Kamm CAD Z 49f zappu: bristle comb

Transliteration eines Teils von VAT 9713

Daneben werden Transliterationen erstellt, also eine Umschrift des Textes in lateinischen Buchstaben, außerdem Übersetzungen und philologische Kommentare.





19 ^d	ḡ ⁸⁸ tukul-[sag]-zag-na ₄	ḥu-tap- <i>pa-lu-ú</i> ⁷	AHW 356 ḥuppālā: Doppelaxt CAD H 263 ḥutpalā: mace ⁵
20	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa ¹ -ṭu	za-ḥa-ṭu- <i>ú</i> ⁷	AHW 1503 zaḥaṭū: eine Streitaxt CAD Z 13 zaḥaṭū: battle-axe
21	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa ¹ -ṭu	qūl-mu- <i>ú</i> ⁷	AHW 927 qulmū: eine Axt CAD Q 299 qulmū: (an ax)
22	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa-ṭu ¹	kal-ma-ak- <i>ru</i> ⁷	AHW 426 kalmak/qrū: eine Streitaxt CAD K 86 kalmakru: battle-ax
23	ḡ ⁸⁸ tukul- <i>ʿ</i> za-ḥa-ṭu ¹	ma-ag-ša- <i>ru</i> ⁷	AHW 577 maḡšaru: sub 2) eine Streitaxt CAD M/1 48 maḡšaru: (an ax)
24	ḡ ⁸⁸ tukul 'kun gar-ra ¹	šá zi-ba-ta [ár]- <i>ru</i> ⁷	AHW 69 armu 1: bedeckt AHW 1524 zibbatu: Schwanz, sub B 2) unkl. in Bez. v Schwert CAD A/2 228 arāmu: mace covered with a "tail" CAD Z 100 zibbatu: tail
25	ḡ ⁸⁸ tukul 'kun si ¹ -ga	'KI-MIN ¹ saḥ- <i>ru</i> ⁷	AHW 1004 saḥāpu: sub 9) überdecken (mit) b) (Waffe) CAD S 31 saḥāpu: to cover "Waffe, die mit einem 'Schwanz' bedeckt ist"
26	ḡ ⁸⁸ tukul niḡ- <i>ʿ</i> sik ¹	[kak]-ki a-za- <i>ru</i> ⁷	AHW 1511 zappu: B ein Kamm CAD Z 49f zappu: bristle comb

Unser Ansatz ist es, online von allen Texten hochauflösende Fotos, die jeweils zoombar sind, anzubieten. Daneben werden Handkopien, die bei schwer lesbaren oder zerstörten Textstellen wie in dem gezeigten Beispiel noch immer unverzichtbar sind, ebenfalls zoombar zur Verfügung gestellt. Die philologischen Bearbeitungen werden als pdf-Datei angeboten, ebenso wie von den Handkopien eine pdf-Version den Nutzern zum individuellen Download zur Verfügung gestellt wird. Der Nutzer kann sich also alle drei Komponenten, Foto, Kopie und Bearbeitung, nebeneinander am Rechner ansehen und die Bearbeitungen gleich am Foto überprüfen.

Projektseite

The screenshot shows the website 'Digitale Keilschrift Bibliothek'. On the left is a navigation menu with sections: 'Suchen', 'Über das Projekt', 'Zusätzliche Texte', 'Transkriptionen', 'Links', 'PDFs', 'Digitale Originale', 'Media', 'Das Team', 'Links', 'FAQ / Kontakt', 'Impressum', and 'Kontakt'. The main content area is titled 'Projektbeschreibung' and contains several paragraphs of text. A photograph of a clay tablet with cuneiform inscriptions is visible on the right side of the page. The text describes the project's goals, the types of tablets included, and the availability of high-resolution photos and transcriptions.

Das Projekt ist online erreichbar unter <http://keil.uni-goettingen.de>.

Problemstellung

- Große Bildmengen
- Im Internet bereitstellen, ohne Verlust von Bildinformationen oder lange Ladezeiten
- Bildmaterial „zoombar“ machen
- Bildmaterial mit Metadaten versehen

Zoomify
Beispiel 1



Auflösung:
20,892 x 20,400 Pixel
Größe: 1,2 GB

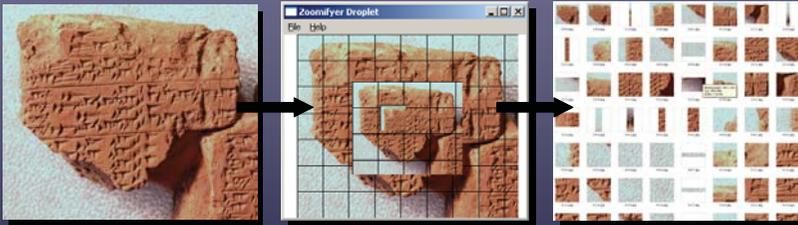
Beispiel einer relativ großen Bilddatei. Die Bilddatei ist trotz einer Dateigröße von 1,2 GigaByte „flüssig“ im Netz zu betrachten. Durch ein kleines Übersichtsbild wird angezeigt, wo man sich in der Gesamtansicht befindet. Dadurch wird das Navigieren vereinfacht und der Überblick bleibt gewahrt.

Bildkonvertierung

- Konvertierung unter MacOSX und Windows möglich
- Konvertierung per bash script und cronjob unter MacOSX
- Containerdatei oder einzelne Dateien
- Beispielkonvertierung

Die Bilddateien müssen für diese Funktionalitäten konvertiert werden.

Konvertierung



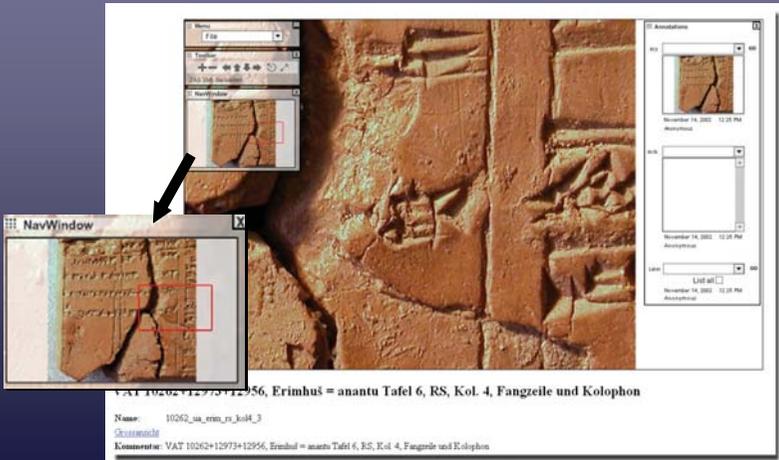
Drag&drop oder Shell Skript

Beispiel

Beispiel einer Bildkonvertierung. Das Gesamtbild wird entweder per drag&drop mit Hilfe eines Applets oder per Kommandozeile umgewandelt. Dabei wird das Bild in eine große Anzahl kleinerer Bildfragmente zerlegt. Zusätzlich wird das Bild noch in verschiedenen Auflösungen gespeichert. Beim Betrachten werden so nur die benötigten Fragmente geladen.

Die Großansicht stellt einige Möglichkeiten zur Verfügung, die eine einfache Bildansicht nicht bieten kann.

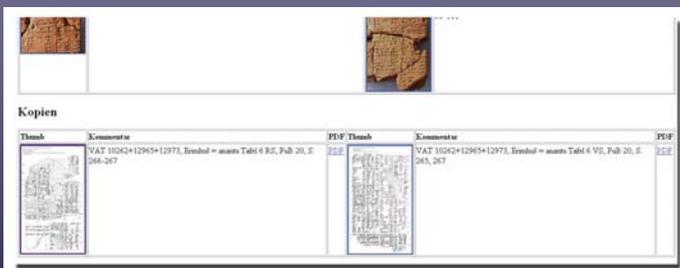
Zoom Tafelbild



Beispiel

Durch die Bildvorschau wird ersichtlich, wo der gezoomte Bildausschnitt in der Gesamtansicht einzuordnen ist.

Kopien der Tafel



Kopien werden auch als Thumbs dargestellt

Für die Tafelkopien werden, genau wie für die Tafelfotos, auch Voransichten generiert und bei den Suchergebnissen angezeigt.

Kopien Tafel



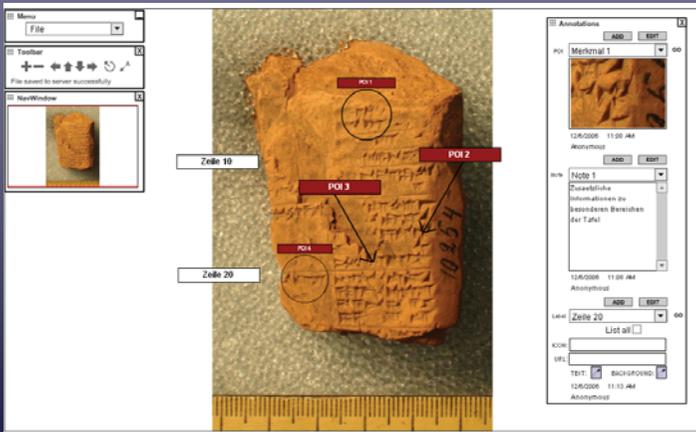
Tafelkopien werden genau wie die Tafelfotos dargestellt

Zusätzliche Bildinformationen werden nicht in das Bild eingefügt



Die Tafelkopien werden auch konvertiert und als zoombare Version veröffentlicht. Im Gegensatz zu den Tafelfotos werden diese aber nicht mit zusätzlichen Metainformationen versehen.

Metadaten



Die Fotos werden mit zusätzlichen Informationen versehen. Man kann bestimmte Regionen markieren, Zeilen nummerieren und zusätzliche Bemerkungen zu bestimmten Bereichen hinzufügen. Dies geschieht, ohne das Foto zu verändern. Es wird dabei eine zusätzliche Ebene über das Foto gelegt, die nach Bedarf auch ausgeblendet werden kann. Das Editieren dieser Informationen erfolgt online und kann vom Bearbeiter jederzeit verändert werden.

Speicherung der Metadaten

- data.xml

```
data.xml
<ZAS>
<POI x="0" y="0" zoom="-1" user="Anonymous" date="20061106132138" id="0" name="Whole Slide" />
<Note text="This is the default note for this POI. Please edit me. I cannot be deleted." user="Anonymous"
date="20061106132138" id="0" name="Default Note" />
<Label x="0.0745" y="0.2784" zoom="19.3806698467815" xscale="90.7034" yscale="90.7034"
user="Anonymous" date="20061106132204" id="0" name="POI 1" movieClip="label_Circle_mc" url=""
textColor="16777215" textBackgroundColor="10040064" poiID="0" />
date="20061106132544" id="4" name="Zeile 10" movieClip="" url="" textColor="0"
textBackgroundColor="16777215" poiID="0" />
...
<Note text="Zusaetzliche Informationen zu besonderen Kennzeichen der Tafel" user="Anonymous"
date="20061106132731" id="1" name="Note 1" poiID="1" />
<POI x="-0.291563883890417" y="0.240090722795138" zoom="120" user="Anonymous"
date="20061106132817" id="2" name="Merkmal 2" />
<Note text="Mehr Informationen" user="Anonymous" date="20061106132830" id="2" name="Note 2" poiID="2"
/>
</ZAS>
```

Diese Daten werden in einer XML-Datei abgespeichert und beim Betrachten des Tafelfotos automatisch nachgeladen. Je nach Möglichkeiten kann auch direkt die XML-Datei editiert werden.

Web 2.0-Techniken für die CDLI

Sebastian Schröder

Max-Planck-Institut für Wissenschaftsgeschichte, Berlin

In Zusammenarbeit mit der Cuneiform Digital Library Initiative (CDLI) in Kalif und mit Unterstützung durch Jörg Kantel und Peter Damerow (MPIWG, Berlin) sowie Dr. Robert Englund und Jacob Dahl (CDLI, Kalifornien).

1. Einleitung

Seit Jahrhunderten haben archäologische Ausgrabungen das Ziel, die Ergebnisse einem breiten wissenschaftlichen als auch öffentlichen Publikum in Form von Büchern, Publikationen oder Ausstellungen zugänglich zu machen.

Leider sind diese Publikationen oder Ausstellungen nur auf das den Autoren oder Veranstaltern zugängliche Material begrenzt. In den meisten Fällen sind die Funde und Ausarbeitungen aber, durch politische oder geografische Grenzen getrennt, nicht ohne größere Hindernisse in Verbindung zu setzen.

Diese Aussage lässt sich am Beispiel des heutigen Irak veranschaulichen. Zurzeit ist hier weder die Besichtigung, Bearbeitung noch die Durchführung von neuen Forschungsprojekten möglich.

Auf Basis dieser Schwierigkeiten bieten neue Techniken und Verhaltensweisen, die das Internet zurzeit unter dem Synonym Web 2.0 vereint, eine hervorragende Möglichkeit, diese Grenzen zu überwinden. Darüber hinaus wird eine neue Art der Publikation geschaffen, die einen weltweiten Informationsaustausch ermöglicht. Es ergibt sich unter anderem der Vorteil, dass ein Peer-Review-Prozess sich nicht über Monate oder Jahre hinzieht, da die Informationen schneller mit anderen im Netz befindlichen Daten verglichen und gegebenenfalls ergänzt sowie diskutiert werden können.

Im Rahmen dieser Arbeit sollten die Anwendungsmöglichkeiten des Web 2.0 respektive Webmapping Tool an dem Beispiel der Ausgrabungstätten von Ur, Uruk und Nimrut evaluiert werden.

- Zunächst wurde überprüft, welche Qualität Geo Coding Tool's (Google Earth/Maps) bieten, die eine Visualisierung erlauben. Auf Basis dieser Erkenntnisse soll eingeschätzt werden, mit welchem Tool die Visualisierung am effektivsten ist und in welchem Maße diese stattfindet. Zusätzlich wurde untersucht, in welchen Datenformaten die Inhalte hinterlegt werden und ob diese auch in lizenzfreie Formate umgewandelt werden können.
- In einem zweiten Schritt wurden die Varianten der Editierung und Verteilung von Geo- und ihren Metadaten mit diesem Tool untersucht.
- Nach Klärung der Datenverteilung wurde die Einbeziehung von dreidimensionalen Objekten und die Berechnung der Polygone und die Größe der Texturen bzw. der Overlay auf den Objekten und auf dem Kartenmaterial untersucht.

Für die Bearbeitung der gestellten Aufgaben wurden einige Beispieldaten vom CDLI-Projekt ausgewählt und testweise implementiert. In diesem Falle wurden die Informationen in einem Filesystem und als Mashup bei Flickr hinterlegt, da zum Projektzeitpunkt kein Zugriff auf die CDLI-Datenbanken bestand.

2. Testumgebungen

Für die Untersuchung wurden eine Browser- (Google Maps) und eine Client-Applikationsvariante (Google Earth) als Testumgebung aufgesetzt. Diese sollen in dem folgenden Abschnitt näher betrachtet werden.

2.1 Google Maps

Zum einen wurde die Google Maps „API“ in ein HTML-Template adaptiert. Durch diese ist es nun möglich, Satellitenkartenmaterial in eine selbst gestal-

tete Webseite einzubinden. Als Koordinaten wurden die zuvor erwähnten Punkte Ur, Uruk und Nimrod in die Testumgebung eingebunden. Die Geodaten wurden in dem Filesystem eines Webservers als KML¹-Dateien abgelegt und können über PHP/Javascript ausgelesen und dargestellt werden. Bei größeren Datenmengen ist auch die Verwendung einer Datenbank vorgesehen.

In den KML-Daten sind unter anderem alle Geo-relevanten Informationen und die Ausgrabungskarten als „overlay“ zusammengefasst. Neben den Satellitenkarten wurde auch eine Arbeitsumgebung für Eintragung, Bearbeitung sowie die Suche von Geo- und Metadaten² eingesetzt. Darüber hinaus ist eine Uploadfunktion der mit Hilfe von Google Earth erstellten KML-Files vorgesehen.

2.2 Google Earth

Als nächstes wurden die gleichen Testdaten in eine Google-Earth-Applikation eingebunden. Vorteil dieser Anwendung ist die maßstabsgerechte Darstellung von Ruinen oder anderen Rekonstruktionen in dreidimensionalen Abbildungen. Die Verteilung der Informationen kann aus Implementierungsgründen nicht wie bei Google Maps über den Webserver erfolgen. Daher wurde ein KML-Loadfile³ generiert, welches auf einen KML-Server⁴ referenziert. Ein Loadfile, integriert in einen Google Earth Client, kann somit die Geo- und Metadaten automatisch und inkrementell von einem Server laden und darstellen

-
1. Keyhole Markup Language, kurz KML, ist ein Austauschformat für die Client-Komponente des Programms Google Earth bzw. Maps. Basiert auf XML 1.0.
 2. Alle projektrelevanten Daten wie Bilder, Karten, Beschreibungen oder Katalognummer der CDLI-Datenbanken.
 3. Ein KML-Load-File enthält nur eine Referenz auf die Daten und Update-Einstellungen. Die Daten verbleiben auf dem Server und werden nur bei Bedarf nachgeladen und im Google-Earth-Speicher gehalten.
 4. Der KML-Server besteht in unserer Anwendung nur aus einem Webserver mit Filesystem und Datenbank. Marker und Koordinaten werden in einem Google-Maps-Mashup eingegeben und in eine Datenbank abgespeichert. Die Geodaten werden mit PHP aus der Datenbank gelesen und als KML in das Filesystem abgelegt oder es werden von einem Google-Earth-Client erzeugte KML-Files in das Filesystem hochgeladen. Zum späteren Zeitpunkt wird ein Script die Load-Files generieren und bereitstellen.

3. Resultate

Nach der Untersuchung der Testanwendungen wurde festgestellt, dass sich Web-2.0-Technologien und -Techniken (Google Earth/Maps) in vollem Umfang für die Bearbeitung von wissenschaftlichen Projekten und deren Visualisierung eignen. Durch den Einsatz von Satelliten- und Überflugkarten ist erstmalig gelungen, Ausgrabungspläne und Fundorte digital für jede Person im Zusammenhang zugänglich zu machen. In dem nächsten Abschnitt werden die Vorteile einer Google-Earth/Maps-Anwendung kurz erläutert.

3.1 Ergebnisse Google Maps

Google Maps ist über die Google API in die Webseite integriert, ein vielfältiges Tool für die Produktion von eigenen Geo-Web-Anwendungen und Services. Durch Unterstützung von Ajax kann man eine fast Desktop-ähnliche Arbeitsumgebung schaffen (Abbildungen 1 und 2), in der nicht nur Ausgrabungspunkte bearbeitet werden können. Ferner lassen sich Anwendungen für alle Arten von Projekten entwickeln, bei denen es einer Verknüpfung bzw. Visualisierung von Geodaten bedarf. In Verbindung mit Wiki- und Blog-Technologien lässt sich auch eine komplette Arbeitsumgebung für geographisch getrennt arbeitende Projektgruppen entwickeln. Die Geodaten werden in einem XML-konformen Format gespeichert (KML/KMZ). Somit ist es möglich, die angesammelten Informationen bei einer Änderung oder Einstellung des KML-Formates von Google in ein anderes XML-Format einzufügen. Diese stehen allen weiteren Prozessen zur Verfügung und es ist somit keine direkte Abhängigkeit von dem Google-Format gegeben.

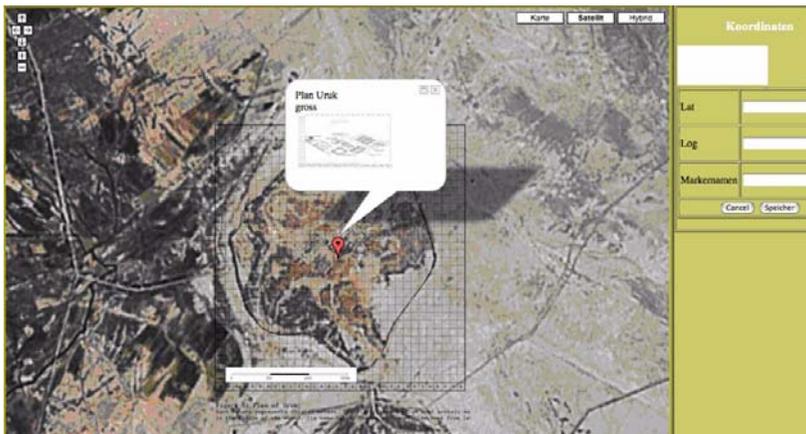


Abb. 1: Testanwendung Google Maps (Grabungsplan Uruk in der Übersicht)

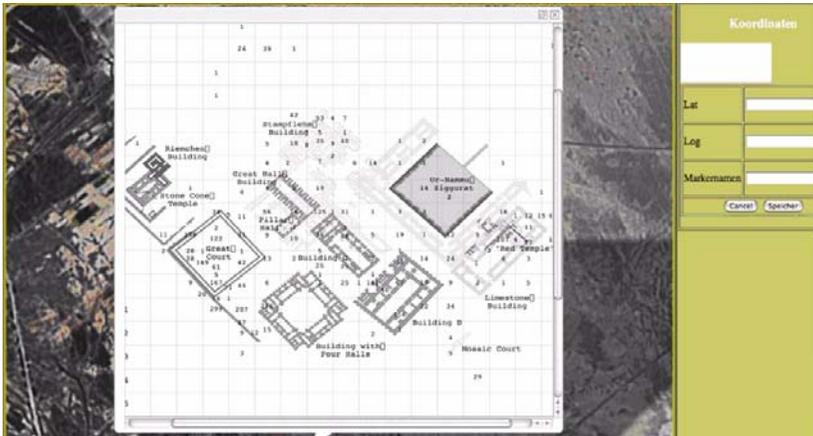


Abb. 2: Testanwendung Google Maps (Grabungsplan Uruk Großprojektion)

3.2 Ergebnisse Google Earth

Google Earth ist im Vergleich zu Google Maps eine Applikation, die auf dem Desktop des Users läuft. Diese kann ebenfalls KML-Files lesen bzw. erzeugen und hat gegenüber Google Maps den Vorteil, dass es eine dreidimensionale Sicht auf das Kartenmaterial ermöglicht (Abbildung 3). In diesem Zusammenhang wird ebenfalls deutlich, dass durch die Implementierung eigener Karten bei schlechten Satellitenbildern durchaus auch eine effektive Visualisierung möglich wird. In Google Maps kann diese nicht realisiert werden, da dort der maximale Zoomfaktor begrenzt ist.

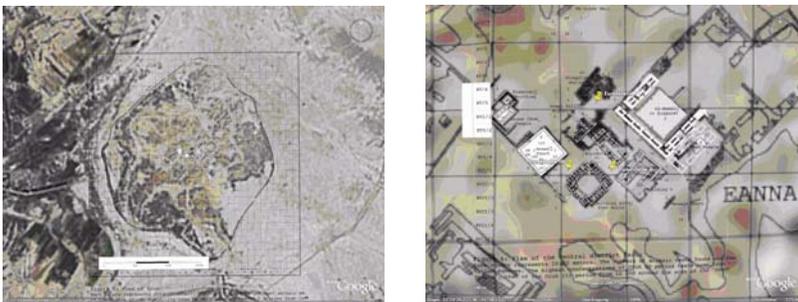


Abb. 3: Markierung von Fundorten in Uruk

Ein weiteres Feature ist die Markierung und Beschreibung von Polygonzügen. Dabei wird ein bestimmter Bereich auf der Karte zusammengefasst und mit Informationen versehen.

In einer Datenbank hinterlegt, können diese Informationen miteinander in Verbindung gesetzt werden. Es besteht die Möglichkeit, über Berechnungen einzelne Geodaten diesen Feldern zuzuordnen bzw. zu bestimmen (Abbildung 4).



Abb. 4: Beispiel für die Bezeichnung bzw. Markierung eines Weges in Nimrud

Darüber hinaus können dreidimensionale Modelle z. B. in Form von Ruinen oder Rekonstruktionen (Abbildung 5) in die Karten eingebunden werden. Bei diesem Prozess werden Polygonmodelle anhand von Grabungsplänen und Rekonstruktionen angefertigt. Diese werden über ein Google-eigenes Tool in den Earth-Client eingebunden und ebenfalls als KML abgespeichert. Der Datenaustausch läuft nach Upload der Daten auf den KML-Server und Generierung des Loadfiles ohne Probleme. Auch die Flickr-Implementierung in der Testanwendung führte zu guten Ergebnissen. Wie in Abbildung 6 zu sehen, ist durch die Kombination von Overlays und 3D-Modellen eine fast perfekte Visualisierung von Ausgrabungsorten möglich. Diese können in Google Earth editiert und als KML abgespeichert werden. Zurzeit ist aber

keine Uploadfunktion von im Client erstellten Geodaten zum eigenen Publishing-Server möglich.

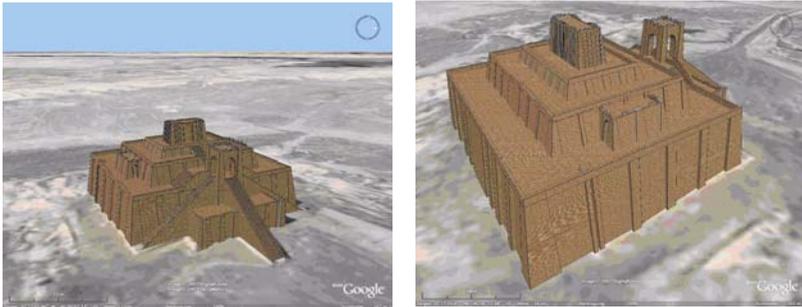


Abb. 5: 3D-Rekonstruktion der Ziggurat von Ur



Abb. 6: Kombierter Einsatz von Overlays, 3D-Rekonstruktionen und Flickr

4. Zusammenfassung

Google Earth und Maps müssen nicht getrennt voneinander betrachtet werden. In Kombination kann man die Stärken der beiden Tools vereinen und eine völlig neue Arbeit und Präsentationsumgebung schaffen (Abbildung 7).

In der Google-Maps-Anwendung werden die Koordinaten anlegt und mit Metadaten und Overlays versehen, da sich Google Maps für die Editierung weitaus besser eignet als Google Earth. Dagegen ist Google Earth ein weitaus besseres Präsentationsmedium, in dem es mit der „Pro“-Version möglich ist, Videos von seinen Kameraflügen über projektrelevante Punkte aufzuzeichnen. Daten, die mit Google Maps auf dem Server als KML-File hinterlegt wurden, können über das Load-File ganz einfach in Google Earth integriert werden.

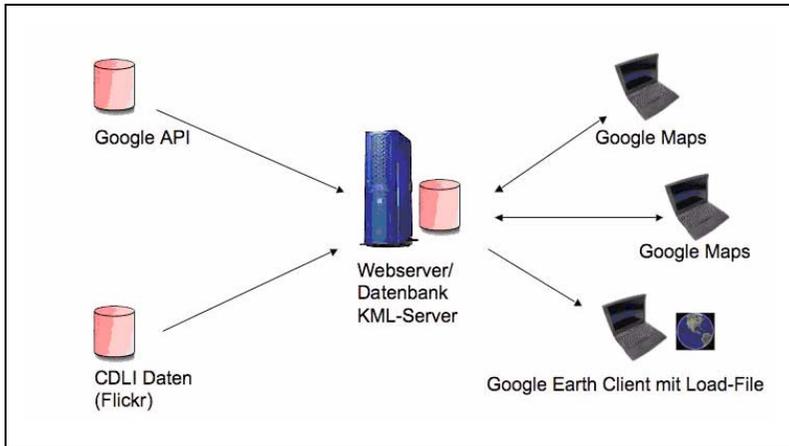


Abb. 7: Aufbau KML-Server-Testanwendung für die CDLI

Als Fragestellung ergeben sich aus diesem Projekt die Punkte der Abhängigkeit von Google Earth/Maps und die Verwendung von Google-eigenem KML-Format als Datenträger.

Google kann seinen Service natürlich ohne weiteres einstellen und die Benutzung von KML-Formaten untersagen. Beim Eintreten dieser, aus heutiger Sicht eher unwahrscheinlichen Situation, besteht die Möglichkeit, einen eigenen Map-Server aufzusetzen und die KML-Files wie oben schon erwähnt in ein anderes XML-Format umzuwandeln. Somit ist die Gefahr einer Abhängigkeit von Google eher gering.

5. Aussichten

Weiterführend zu diesem Projekt könnte man diese Arbeitsoberfläche für aktuelle Ausgrabungen verwenden, in dem man sich die Google Earth Enterprise Edition zu nutze macht.

Diese bringt eine Integration von GPS-Geräten und -Empfängern mit sich, durch die es möglich ist, in der Ausgrabung sofort die Fundstellen mit Geodaten zu verknüpfen und diese anschließend einfach als KML-File auf einen Server zu laden. Somit können diese Daten in wenigen Minuten, an allen Orten der Welt verfügbar gemacht werden. Über ein Loadfile dieses Projektes kann man die Fortschritte in Google Earth beobachten und parallel die Ausgrabungsfunde (-stellen) diskutieren, zuordnen oder übersetzen. Verbunden mit Wiki und Blog-Technologie wäre es denkbar, die Bearbeitungszeit von Ausgrabungen oder Ähnlichem auf ein Minimum zu reduzieren, da viele Prozesse wie Archivierung, Zuordnung, Vergleich mit anderen Fundstücken und Visualisierung schon während der Ausgrabung parallel ablaufen könnten.

Weitere Vorteile sind die lokale Speicherung von Google-Karten auf dem Server. Diese können durch eigene Satelliten- oder Überflugkarten erweitert und ergänzt werden. Des Weiteren erfolgt die Bearbeitung, Speicherung von KML-Daten und die Verknüpfung von Google-Maps/Earth-Anwendungen auf einem Produktionsserver. Zusätzlich dazu können von Google geladene Karten gesichert und im Falle einer Löschung oder Verschlechterung durch ein Update (wie bereits geschehen bei Ur im Irak) wieder zurückgespielt werden.

Natürlich ist dieser Workflow nicht auf Ausgrabungen beschränkt. Er lässt sich auch für andere Projekte verwenden, die sich mit Kartenmaterial befassen bzw. sich in ihren Arbeiten darauf beziehen.

Zukünftige HPC-Systeme der GWDG

Oswald Haan

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen

1. Einleitung: High Performance Computing (HPC) bei der GWDG

Der Einsatz von leistungsfähigen Rechnern zur Simulation komplexer Modelle und zur Analyse experimenteller Befunde gehört zum methodischen Grundgerüst der naturwissenschaftlichen Grundlagenforschung. Aufgabe der GWDG ist es deshalb, dem Stand der Technik entsprechende Hochleistungsrechnersysteme bereitzustellen, die den Wissenschaftlerinnen und Wissenschaftlern der Universität Göttingen und der Max-Planck-Gesellschaft eine wesentliche Grundlage zur Erzielung international anerkannter Forschungsleistungen bietet.

High Performance Computing (HPC) bei der GWDG

Einsatz von HPC als Forschungswerkzeug in Naturwissenschaft und Technik:

Theorie: Simulation komplexer Modelle

Experiment: Bearbeitung und Analyse großer Datensätze

Ressourcen:

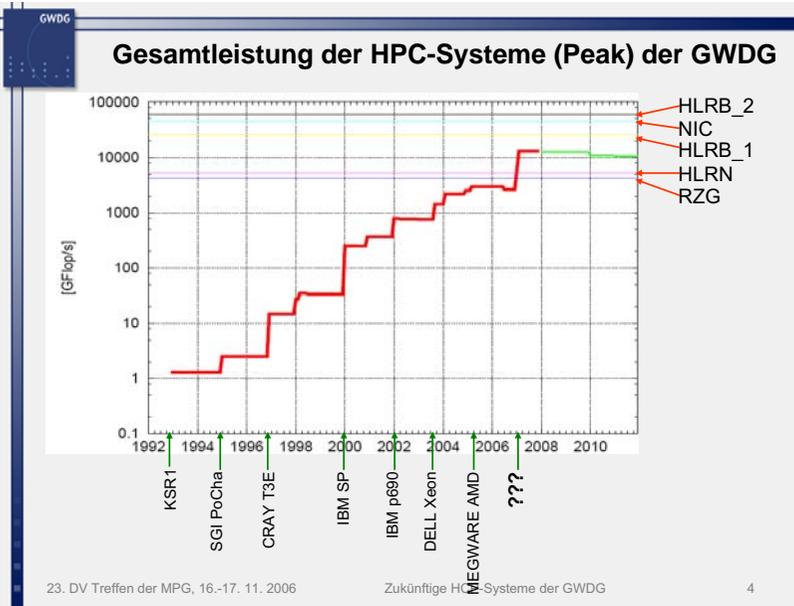
Arbeitsplatz- und Institutsrechner

HPC-Ressourcen des lokalen Rechenzentrums (GWDG)

Überregionale Höchstleistungsrechner (RZG, HLRN, HLRS, HLRB, NIC)

Wegen der rasanten technologischen Entwicklung steigert sich die Leistungsfähigkeit der Hochleistungsrechner bei konstantem Preis um einen Faktor 10 innerhalb von vier bis sechs Jahren, so dass die Betriebsdauer eines Hochleistungsrechners i. d. R. etwa fünf Jahre betragen sollte und in entsprechenden Zeitabständen Systeme der neuesten Technologie beschafft werden müssen.

Die GWDG hat seit 1993 einen dem Technologiestand und den Finanzierungsmöglichkeiten ihrer beiden Gesellschafter angepassten Bestand an Hochleistungsrechnern beschafft und betrieben. In der Folie 4 ist der exponentielle Leistungszuwachs der von der GWDG betriebenen HPC-Systeme in der logarithmischen Darstellung als linearer Anstieg sichtbar. Eingetragen ist auch schon die Leistungserhöhung durch die neuen Systeme SGI Altix 4700 und MEGWARE Woodcrest-Cluster, die die Gesamtleistung wieder auf den bisher eingehaltenen Wachstumspfad bringen. Ebenso ist aus der Extrapolation der Gesamtleistung für die kommenden Jahre abzulesen, dass ab 2008 wieder Bedarf an Erweiterung der Rechenleistung ansteht, wenn die bisherige Wachstumsgeschwindigkeit beibehalten werden soll.



2. Das Beschaffungsverfahren

Das Beschaffungsvorhaben wurde mit einer im Herbst 2005 durchgeführten Benutzerbefragung vorbereitet, durch die der Bedarf an Hochleistungsrechnerkapazität für die kommenden Jahre ermittelt wurde. In den Folien 5, 6 und 7 sind die Ergebnisse dieser Befragung zusammengefasst:

Bedarfslage

Benutzerumfrage Ende 2005: HPC-Bedarf gemeldet für

39 Projekte von 16 Forschungsgruppen aus 10 Instituten der Universität Göttingen:

Astrophysik, I. Physik, Theoretische Physik, Geophysik, Materialphysik, Anorganische Chemie, Organische u. biomolekulare Chemie, Physikalische Chemie, Geowissenschaften, Bioklimatologie

21 Projekte von 9 Forschungsgruppen aus 3 MPIen im Göttinger Raum:

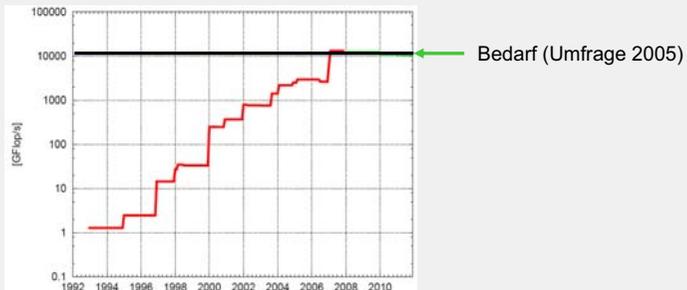
Sonnensystemforschung, Dynamik u. Selbstorganisation, Biophysikalische Chemie

Bedarfslage

Benutzerumfrage Ende 2005: im nächsten Jahr benötigte Leistung

Universität: 5,23 TFlop/s Peak

MPI: 6,05 TFlop/s Peak



Bedarfslage

Benutzerumfrage Ende 2005: Anwendungsspektrum

	Gesamtbedarf [TFlop/s] Peak	Message Passing	Shared Memory
Universität	5,23	78 %	22 %
MPI	6,05	66 %	34 %

Die jüngste Neubeschaffung der Parallelrechnersysteme bei der GW DG wurde aus Investitionsmitteln der Gesellschafter für die Jahre 2006 und 2007 sowie aus Investitionsmitteln der Abteilung „Physik der Planeten und Kometen“ von Prof. Christensen des Max-Planck-Instituts für Sonnensystemforschung in Katlenburg-Lindau finanziert.

Finanzrahmen

Gemeinsame Beschaffung von GWDG und
Max-Planck-Institut für Sonnensystemforschung (MPS)

GWGD Investitionsmittel	2 M € (verteilt auf 2006 u. 2007)
davon Universität Göttingen	1 M € (HBFG-fähig)
MPG	1 M €
MPS Berufungsmittel	0,6 M €

Die Benutzerbefragung gab auch Aufschluss über Programmiermodelle und Kommunikationsanforderungen der geplanten Anwendungen. Insbesondere wurde deutlich, dass die günstigste Konfiguration für das ermittelte Anwendungsprofil eine Kombination aus unterschiedlichen Architekturen ist. Für OpenMP-Programme und kommunikationsintensive Anwendungen wird eine Shared-Memory-Komponente benötigt. Message-Passing-Programme mit mittlerem und geringem Kommunikationsbedarf können kostengünstiger mit einer Cluster-Komponente bearbeitet werden.

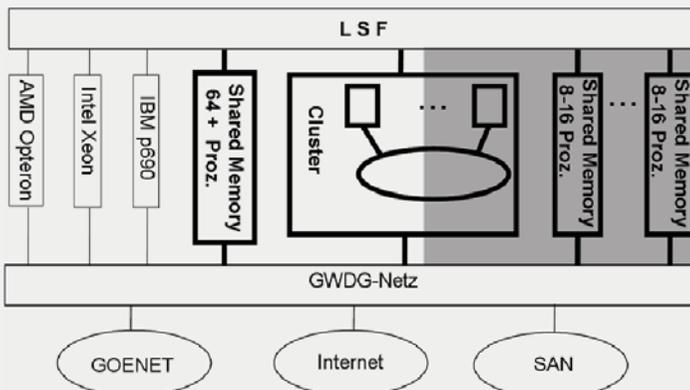
Entsprechend dieser Analyse wurde im Frühjahr 2006 ein Beschaffungskonzept erstellt, das dem Beratenden Ausschuss für Rechenanlagen in der Max-Planck-Gesellschaft (BAR) und der Kommission für Rechenanlagen der Deutschen Forschungsgemeinschaft (KfR) zur Begutachtung vorgelegt und von diesen befürwortet wurde. Aufgrund der Befürwortung durch die KfR wurde der Finanzierungsanteil des Gesellschafters Universität Göttingen in das Programm des Hochschulbauförderungsgesetzes aufgenommen, wodurch dem Land Niedersachsen 50 % dieser Mittel vom Bund erstattet werden.

Der ursprüngliche Zeitplan sah die Beschaffung der beiden Komponenten in zwei Phasen vor.

Beschaffungsplanung

- Phase 1: SMP-Komponente, min. 64 Prozessoren pro Knoten
gemeinsam mit MPS
Cluster-Komponente, min. 2 Prozessoren pro Knoten,
Hochleistungs-Kommunikationsnetz
- Phase 2: SMP-Komponente, min. 16 Prozessoren pro Knoten
Erweiterung Cluster-Komponente
- Zeitplan: Phase 1 bis Ende 2006
Phase 2 je nach Freigabe der Mittel im Jahr 2007

Beschaffungsplanung



Wegen der Neuordnung des HBFVG-Verfahrens wurde der GWDG nahegelegt, beide Phasen im Jahr 2006 abzuwickeln. Dies spiegelt sich im zeitlichen Ablauf des Beschaffungsvorhabens wieder.

GWDG	
Beschaffungsablauf	
Sept/Okt 2005	Bedarfserhebung
Apr 2006	HBFG-Anmeldung für Invest-Anteil der Uni
19. 07. 2006	Veröffentlichung der Vergabebekanntmachung
11. 08. 2006	insgesamt 14 Firmen erhalten Verdingungsunterlagen
16. 08. 2006	DFG Empfehlung zur Aufnahme in HBFG Rahmenplan bis 0,9 M € (10% Reduktion)
19. 09. 2006	Einspruch gegen Reduktion erhoben
27. 09. 2006	Eingang von 3 Angeboten für SMP-Komponente und 6 Angeboten für Cluster-Komponente
04. 10. 2006	Auswahl: SGI für SMP-Komponente MEGWARE für Cluster-Komponente
31. 10. 2006	Vertragsabschluss mit MEGWARE
07. 11. 2006	Vertragsabschluss mit SGI
08. 12. 2006	Lieferung SMP-Komponente (Phase1+Phase2)
28. 12. 2006	Lieferung Cluster-Komponente (Phase1+Phase2)
23. DV Treffen der MPG, 16.-17. 11. 2006	Zukünftige HCP-Systeme der GWDG
	11

Im Sommer 2006 erfolgte die europaweite Ausschreibung für ein HPC-System, bestehend aus einer Shared-Memory-Komponente mit mindestens 64 Prozessor-Kernen pro Systemknoten und einer Cluster-Komponente mit einem leistungsfähigen Kommunikationsnetz. Es gingen Angebote von insgesamt acht Herstellern ein, sechs für die Cluster-Komponente, drei für die Shared-Memory-Komponente (nur ein Hersteller gab ein Angebot für beide Komponenten ab).

Das Hauptkriterium für die Auswahl war die für das feststehende Investitionsvolumen angebotene Leistung, die durch vorgegebene Benchmarks ermittelt wurde. Der HPCC-Benchmark gab Auskunft über Basisdaten des angebotenen Systems (CPU-Leistung, Speicherbandbreite und Kommunikationsleistung), Anwendungen wie CPMD, Gaussian, FFTW oder MAGIC über die real auf den Systemen erzielbare Anwendungsleistung.

Bewertungskriterien

Hauptkriterium: höchste Leistung für vorgegebenen Preis
Leistungsnachweis durch Benchmarks

HPC Challenge Benchmark:

STAR-DGEMM	CPU-Leistung
STAR-STREAMTRIAD	Speicherzugriff
PINGPONG	Punkt-zu-Punkt-Kommunikation
RANDOMRING	gleichzeitige paarweise Komm.
TRANSDOSE	globale Kommunikation

Anwendungsbenchmarks:

MAGIC	Magnetohydrodynamic
CPMD	Car-Parinello Molekulardynamik
GAUSSIAN	Quantenchemie
FFT	Fourier-Transformation

Einige Ergebnisse der abgegebenen Benchmarks sind in den folgenden beiden Folien zusammengefasst:

Benchmark-Ergebnisse Einzelprozessor

	Intel Itanium 2 Montecito (1.6 GHz)	IBM Power5+ (2.1 GHz)	Intel Xeon Woodcrest (3.0 GHz)	AMD Opteron (2.6 GHz)
RDGEMM [GFlop/s]	6.14	8.52	9.42	4.76
STREAM Triade [GB/s]	2.63	3.49	1.97	3.23

Benchmark-Ergebnisse Mehrprozessor

	Altix 4700 Intel Itanium 2 Montecito (1.6 GHz)	pSeries 595 IBM Power5+ (2.1 GHz)	Infiniband Intel Xeon Woodcrest (3.0 GHz)	Infiniband AMD Opteron (2.6 GHz)
PTRANS auf 64 cores [GB/s]	13.20	6.77	6.96	8.77
FFT auf 64 cores [GFlop/s]	14.46	26.39	13.15	16.17
CPMD auf 32 cores [sec]	353	230	587	767
MAGIC auf 64 cores [sec]	1.15	0.96		

Auf der Basis dieser Benchmarks war das Angebot der Firma SGI für die Shared-Memory-Komponente das bei weitem günstigste. Für die Cluster-Komponente lag die Firma MEGWARE vorne, allerdings mit nicht so großem Abstand zum zweitbesten Anbieter.

Gesamtbewertung

SGI Altix 4700 mit Montecito Prozessoren:

mehr als doppelt so hohe Gesamtleistung wie Mitbewerber

MEGWARE Infiniband Cluster mit Woodcrest Prozessoren:

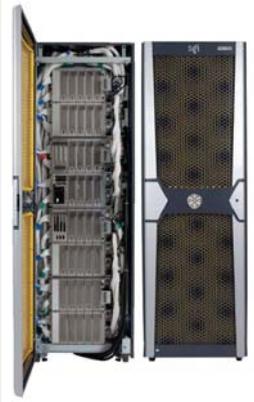
Mehr als 5% höhere Gesamtleistung wie Mitbewerber

3. Beschreibung der neuen Systeme

Die folgenden Folien geben Auskunft über Technik und Konfiguration des ausgewählten Shared-Memory-Systems:

GWDC

SMP-Komponente: SGI Altix4700

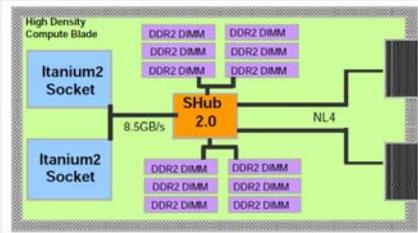


- 256 Dual-Core Itanium „Montecito“ 1,6GHz, 18MB L3 Cache
- 1280GB RAM
- Zunächst 3,6TB lokaler Scratch-Bereich
- SAN-Anschluss
- Wassergekühlt
- Nutzbar als eine SMP-Partition unter SuSE-Linux
- 3,3Tflops Peak-Leistung

23. DV Treffen der MPG, 16.-17. 11. 2006 Zukünftige HCP-Systeme der GWDC 16

GWDC

Altix 4700 High Density Compute Blades

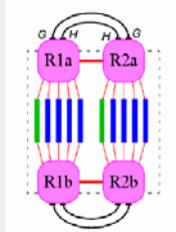


- 2 Dual-Core CPU's
- 8GB Memory
- 2 NUMALink4 Verbindungen

23. DV Treffen der MPG, 16.-17. 11. 2006 Zukünftige HCP-Systeme der GWDC 17

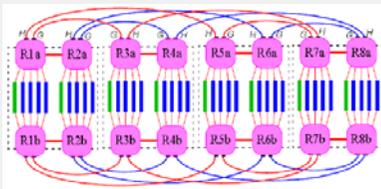
Einzelnes Blade Chassis (IRU)

Maximal 10 Blades, verbunden durch 4 NumaLink Router mit je 8 Ports

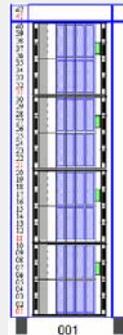
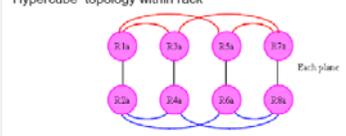


Volles Rack mit 4 IRUs

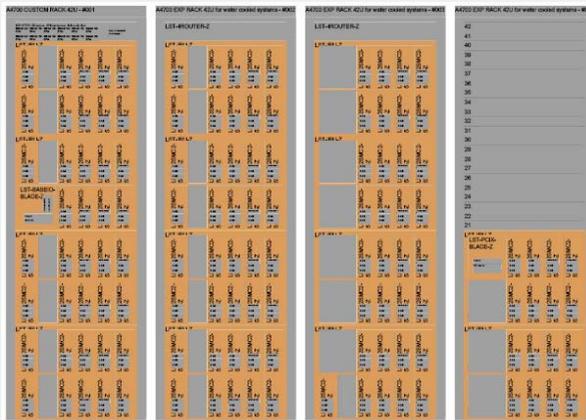
Maximal 40 Blades, verbunden durch 16 NumaLink Router



Hypercube topology within rack



GWDG-System: 14 IRUs in 4 Racks



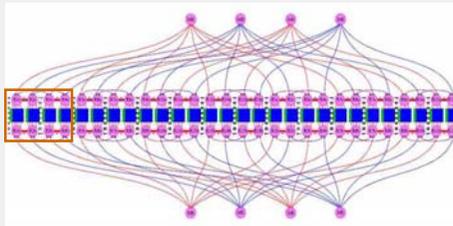
23. DV Treffen der MPG, 16.-17. 11. 2006

Zukünftige HCP-Systeme der GWDG

20

NUMALink4 Konfiguration des GWDG-Systems

64 NUMALink4 Router, davon 8 Metarouter



Non Uniform Memory Access (NUMA) Architektur bedeutet unterschiedlich lange Kommunikationswege:

- Innerhalb der Kleingruppen (von Blade zu Blade) : 1 Router
- Innerhalb der Gruppen (von Kleingruppe zu Kleingruppe) : 2 Router
- Innerhalb des Gesamtsystems (von Gruppe zu Gruppe) : 3 Router

23. DV Treffen der MPG, 16.-17. 11. 2006

Zukünftige HCP-Systeme der GWDG

21

Das ausgewählte Cluster-System wird in den folgenden Folien beschrieben:

GWDC

Cluster-System: MEGWARE

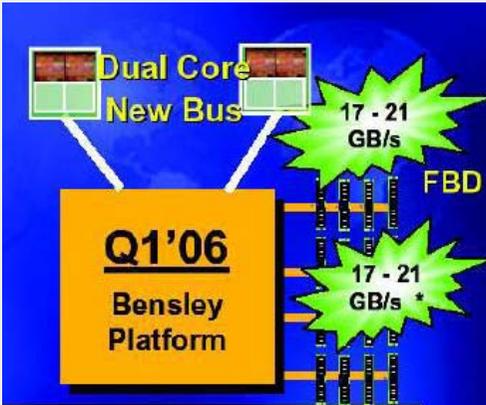


- 150 Knoten mit je 2 Xeon „Woodcrest“ (Dual-Core), 8GB RAM, 400GB HD
- Insgesamt 600 Cores, 1,2TB RAM, 7,2Tflops Peak-Leistung
- Infiniband Highspeed Interconnect, SAN-Anschluss
- Wasserkühlung

23. DV Treffen der MPG, 16.-17. 11. 2006 Zukünftige HCP-Systeme der GWDC 22

GWDC

MEGWARE Zwei-Prozessor-Knoten



The diagram illustrates a node architecture with a central orange box labeled "Q1'06 Bensley Platform". Above it, two green boxes represent "Dual Core" processors connected via a "New Bus". To the right, a vertical stack of components is shown, with two green starburst callouts indicating a bandwidth of "17 - 21 GB/s" for the "FBD" (Front Board) connections.

23. DV Treffen der MPG, 16.-17. 11. 2006 Zukünftige HCP-Systeme der GWDC 23

Infiniband High Speed Interconnect

Mellanox Host Channel Adapter (HCA): MHES14-XTC
Single-Port 10Gb/s PCI Express x4 MemFree Channel Adapter



Flextronics 144 Port Modular InfiniBand switch

10 Gb/s bidirektionale Bandbreite für 144 Ports
2.88 Tb/s aggregierte Bandbreite



4. Zusammenfassung

Die wesentlichen Daten der zukünftigen HPC-Systeme der GWDG sind hier noch einmal zusammengefasst:

Zusammenfassung

- **Neue HPC-Systeme der GWDG**
 - SMP System SGI Altix 4700:
 - 512 Cores Montecito
 - 3,3 TFlop/s Peak,
 - 1,2 TB gemeinsamer Speicher
 - Cluster System MEGWARE
 - 600 Cores Woodcrest,
 - 7,2 TFlop/s Peak,
 - 1,2 TB verteilter Speicher
- Inbetriebnahme 1. Q. 2007

Ein System zur Bearbeitung weltweiter Bewerbungen auf Forschungsstellen der Max-Planck-Gesellschaft

Wilfried Grieger

Gesellschaft für wissenschaftliche Datenverarbeitung mbH Göttingen

1. Ausgangslage

Das Präsidium der Max-Planck-Gesellschaft hatte erstmalig im September 2004 beschlossen, ein Bewerbungsprogramm für Leiter selbstständiger Nachwuchsgruppen innerhalb der MPG auszuschreiben. Die Ausschreibung sollte weltweit erfolgen, wobei zwischen 500 und 1.500 Bewerberinnen und Bewerber erwartet wurden. Der Bewerbungszeitraum sollte vom 20.10. bis zum 02.12.2004 dauern.

Da solch eine hohe Zahl von Bewerbungen, die nach klassischem Verfahren per E-Mail oder per Post bearbeitet wird, lediglich durch einen extrem großen personellen Aufwand zu bewältigen ist, sollte die Bearbeitung der eingegangenen Bewerbungen vollständig elektronisch erfolgen.

Die GWDG hatte daraufhin den Auftrag erhalten, ein entsprechendes elektronisches Bewerbungsverfahren zur Verfügung zu stellen. Auf Grund der

bisherigen positiven Erfahrungen mit Datenbanken unter Lotus Software wurde das Verfahren mit Lotus Notes der Firma IBM realisiert.

2. Anforderungen an das elektronische Bewerbungsverfahren

Auf Grund der weltweiten Ausschreibung und der heterogenen Systeme, auf denen die Wissenschaftlerinnen und Wissenschaftler arbeiten, sollten die Bewerbungsunterlagen vollständig über einen WWW-Browser einreichbar sein. Die Bewerbungsunterlagen sollten dabei aus drei Teilen bestehen:

- eine Anmeldung (Registration), die alle persönlichen Daten der Bewerberin oder des Bewerbers und das ausgesuchte Max-Planck-Institut enthält, an dem die selbstständige Nachwuchsgruppe gegründet werden soll
- eine ausführliche Beschreibung des vorgeschlagenen Forschungsprojekts (Application)
- mindestens zwei Empfehlungsschreiben (Letter of Recommendation) von Wissenschaftlerinnen oder Wissenschaftlern, die das Forschungsprojekt unterstützen

Die nacheinander eingereichten Dokumente sollten automatisch der richtigen Bewerberin oder dem richtigen Bewerber zugeordnet werden.

Auch die Gutachterinnen und Gutachter sollten per WWW-Browser auf die eingereichten Bewerbungsdaten zugreifen können.

3. Registration-Code

Die automatische Zuordnung der Dokumente zur richtigen Bewerberin oder zum richtigen Bewerber wurde durch die Vergabe eines eindeutigen Registration-Codes gelöst. Dieser Code wurde bei der Anmeldung (Registration) der Bewerberin oder dem Bewerber sowohl per Bildschirm als auch per E-Mail mitgeteilt.

Bei der Eingabe der Application und der Letter of Recommendation wurde dieser Code angefordert. Insbesondere musste also die Bewerberin oder der Bewerber den Registration-Code an die Wissenschaftlerinnen oder Wissenschaftler weitergeben, die einen Letter of Recommendation verfassen sollten.

4. Registration

Die Anmeldung zur Teilnahme am Bewerbungsverfahren erfolgte über die Registration-Maske, die im oberen Teil die persönlichen Daten der Bewerberin oder des Bewerbers anfordert:

**Selbstständige Nachwuchsgruppen
(Independent Max Planck Junior Research Groups)**

Registration Form

Please fill in **each** of the following fields:

Last Name:
(please without titles)

First Name:

Date of Birth:
(please select year and month) Year: Month:

Home Institution:
(Institute and Country sufficient)

Your Nationality:

Email Address:

Email Address verification:

Title of proposed Talk:

Im unteren Teil der Registration-Maske erfolgt die Auswahl derjenigen Max-Planck-Institute, in denen die selbstständigen Nachwuchsgruppen etabliert werden sollen:

Registration for Section:

Biology and Medicine (BM)
 Chemistry, Physics and Technology (CPT)
 Cultural, Social and Human Sciences or Legal Studies (H)

If you are not sure, please decide for the Section to which your first-priority Max Planck Institute belongs (see below).

Please note: In the BM Section (only) we offer **alternatively** to the free selection registration several institutional positions at **two given MPIs** and the **Friedrich-Miescher-Laboratory** in Tübingen (see announcements).

Free selection Registration:
Please choose up to three MPIs in which you would like to perform your research project:
(Please note: The **first preference** (but only this) MPI **must** belong to the Section for which you register)

	Section BM Please select *	Section CPT Please select *	Section H Please select *
1st priority (obligatory)	<input type="text"/>	<input type="text"/>	<input type="text"/>
2nd priority (optional)	<input type="text"/>	<input type="text"/>	<input type="text"/>
3rd priority (optional)	<input type="text"/>	<input type="text"/>	<input type="text"/>

Institute Registration (BMS only):
Please choose the MPI in which you want to apply:

*Dash (-) = no institute selected

Nach dem Absenden der vollständig ausgefüllten Eingabemaske wird am Bildschirm unter anderem der automatisch vergebene, eindeutige Registration-Code angezeigt, der einige Minuten später auch per E-Mail an die Bewerberin oder den Bewerber geschickt wird.

5. Application

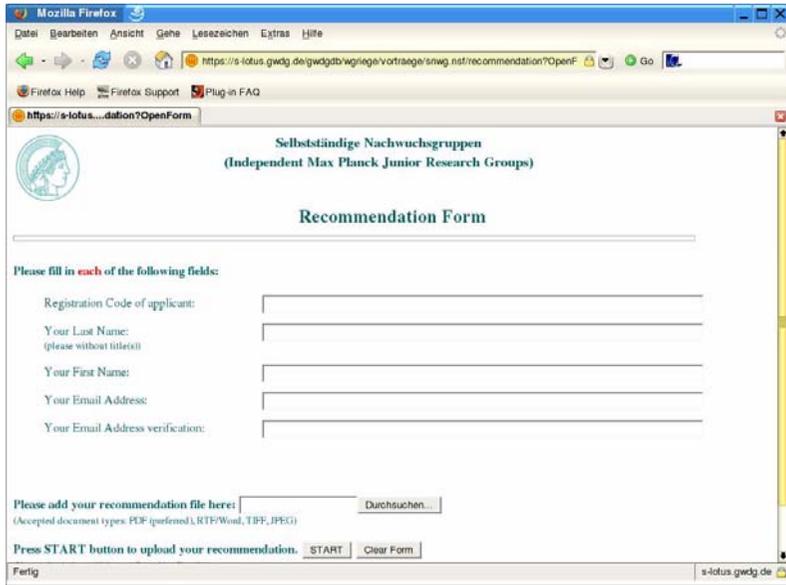
Die Beschreibung des eigentlichen Forschungsprojekts wird über die Application-Maske eingereicht, und zwar als PDF-File. Wie oben bereits erwähnt, ist zusätzlich die Eingabe des Registration-Codes erforderlich:

The screenshot shows a web browser window titled "Application - Mozilla Firefox". The address bar contains the URL: <https://s-lotus.gwdg.de/gwdgdb/wsp/leje/vortraege/snwg.nsf/application?OpenForm>. The page content includes:

- A logo of a woman's head in a circular frame on the left.
- The title "Selbstständige Nachwuchsgruppen (Independent Max Planck Junior Research Groups)" centered at the top.
- The heading "Application Form" centered below the title.
- A section "Please fill in the following field:" containing a text input field labeled "Registration Code:".
- A section "Please add your Application PDF file here:" with a file selection button labeled "Durchsuchen...".
- A "Please note:" section stating: "Some internet browsers have problems with transfer of very large files. Thus, if your PDF file exceeds 10 MB, you may consider splitting it up. In this - and only this - case your registration code can be used for two subsequent submissions; otherwise it can be used only once."
- A "Press START button to upload your application." section with "START" and "Clear Form" buttons. A note below reads: "(Your submission will be confirmed by Email)".
- A red warning line: "Please note that you can not change your data once they are submitted!".
- A status bar at the bottom with the text "Fertig" on the left and "s-lotus.gwdg.de" on the right.

6. Letter of Recommendation

Die Letter of Recommendation werden zusammen mit dem Registrierungscode über die folgende Maske eingereicht:



The screenshot shows a Mozilla Firefox browser window displaying a web page for the "Selbstständige Nachwuchsgruppen (Independent Max Planck Junior Research Groups)". The page title is "Recommendation Form". The form contains the following fields and buttons:

- Registration Code of applicant:
- Your Last Name: (please without titles):
- Your First Name:
- Your Email Address:
- Your Email Address verification:
- Please add your recommendation file here:
- Accepted document types: PDF, pdfencoded, RTF, Word, TBF, JPEG
- Press START button to upload your recommendation.

The status bar at the bottom of the browser window shows "Fertig" and the URL "s-lotus.gwdg.de".

7. WWW-Browser

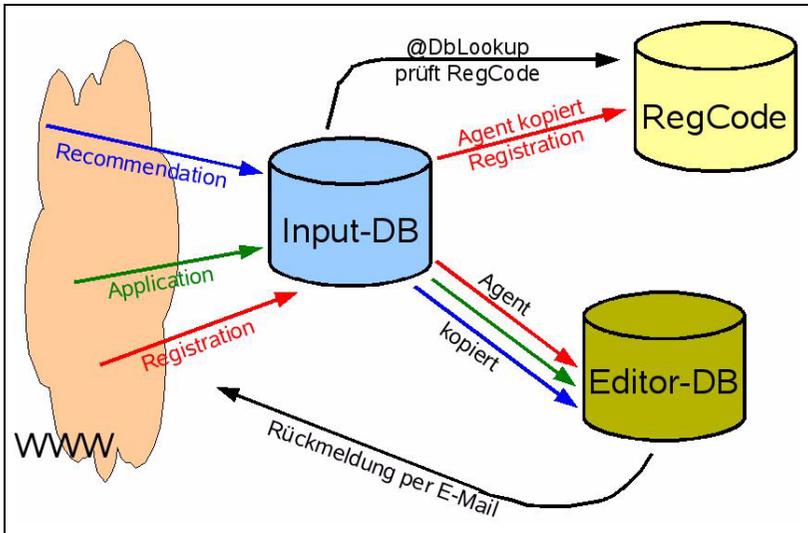
Für den Zugang zur Lotus-Datenbank über das WWW wurden die folgenden Browser erfolgreich getestet:

- MS Internet Explorer
- Mozilla Firefox
- Mozilla
- Netscape
- Opera
- Konqueror
- Safari

Verwendet wurden dabei die letzten Versionen der Browser und dafür auch, soweit verfügbar, unterschiedliche Betriebssysteme.

8. Architektur des Systems unter Lotus

Das folgende Bild soll die Architektur des gesamten Bewerbungssystems unter Lotus-Software verdeutlichen:



Das System besteht dabei aus drei verschiedenen Datenbanken: Die Input-DB nimmt alle Eingaben, die die Bewerberinnen und Bewerber betreffen, auf. Automatisch werden alle Eingaben in die Editor-DB kopiert. Alle Registrierungen landen ebenfalls automatisch in der Datenbank RegCode. Die Rückmeldungen per E-Mail an die Bewerberinnen und Bewerber erfolgt automatisch aus der Editor-DB heraus.

9. Begutachtung

Die eingegebenen Daten wurden von einem Mitarbeiterstab in der Generalverwaltung der MPG auf Vollständigkeit und Lesbarkeit kontrolliert. Diese Kontrolle erfolgte jedoch nicht über einen WWW-Browser, sondern über den Notes Client, der gegenüber dem Browser den Vorteil hat, die Bearbeitung effizienter und einfacher durchführen zu können. Insbesondere ist die Zugriffssicherheit über den Client gegenüber dem Browser deutlich verbessert.

Bei festgestellten Mängeln in den eingereichten Daten startete eine Mitarbeiterin oder ein Mitarbeiter in der Generalverwaltung ein Programm (Agent), das die Mängel per E-Mail an die Bewerberin oder den Bewerber teilt.

Die Auswahl der Bewerberinnen oder Bewerber erfolgte durch eine Reihe von Gutachterinnen und Gutachter, die per WWW-Browser auf die Daten in der Editor-DB zugriffen, natürlich Passwort-geschützt und ssl-verschlüsselt. Dabei waren für jede Sektion lediglich die Bewerbungen für diese Sektion sichtbar:



Eine Bewerberliste hatte dann etwa folgendes Aussehen:

#	Name	Type	Date of Birth	First Priority MPI	Second Priority MPI	Fixed Institutes	Title of Talk
1	Bush, George W.						
2	Merkel, Angela						
2.1		Registration	05.19.56	MPI Biophysics	MPI Physics		My visions
2.2		Recommendation					
2.3		Recommendation					
2.4		Application					
3	Schroeder, Gerhard						

Nach dem Anklicken eines Letters of Recommendation konnte dann auch der Inhalt des Empfehlungsschreibens angezeigt werden:

Selbstständige Nachwuchsgruppen
(Independent Max Planck Junior Research Groups)

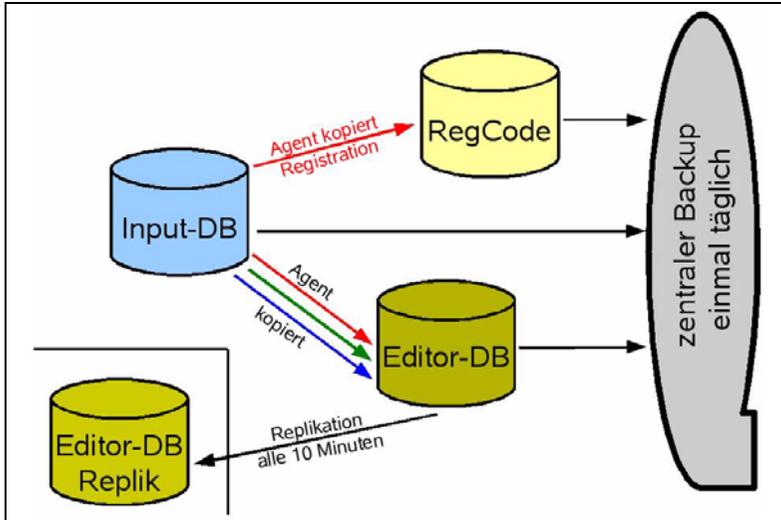
Recommendation Letter

Name of applicant: Merkel, Angela
Name of referee: Stoiber, Edmund
Email Address of referee: edmund@stoiber.de

 [unix.pdf](#)

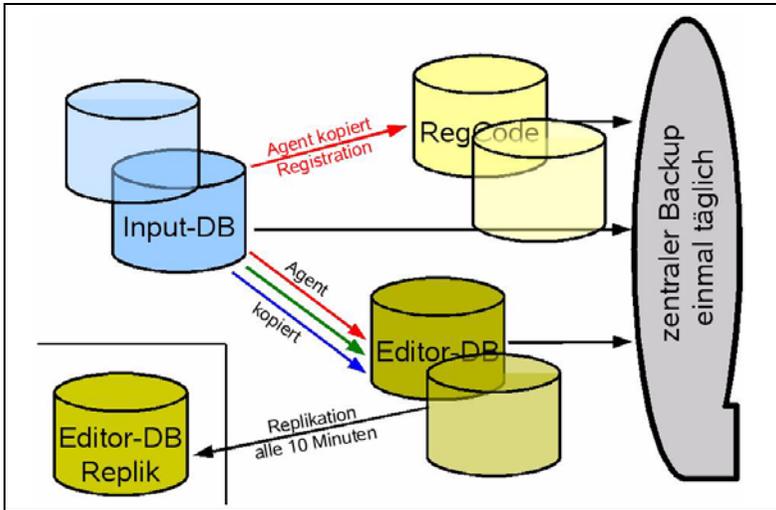
10. Datensicherheit

Im gesamten Bewerbungsverfahren wurde selbstverständlich auf ein hohes Maß an Datensicherheit Wert gelegt. Das folgende Schaubild soll die dafür eingesetzten Maßnahmen erläutern:



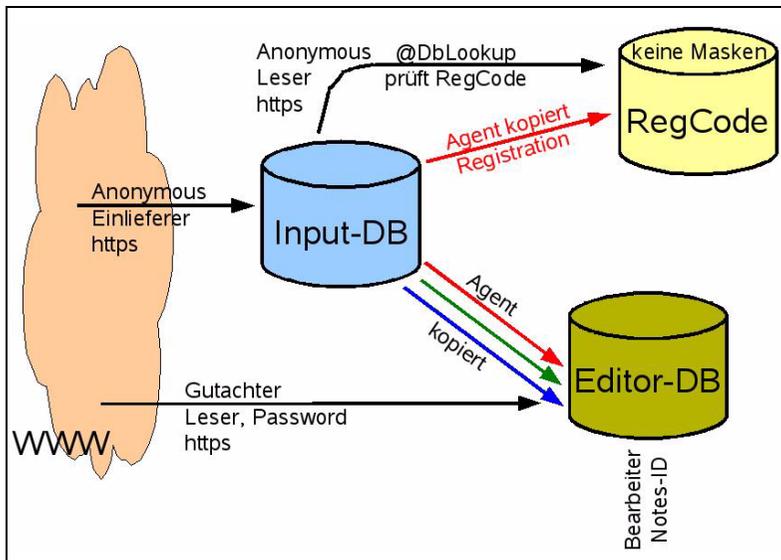
Üblicherweise werden alle Datenbanken einmal täglich durch den zentralen Backup der GWDG auf Magnetbänder kopiert. Da diese Sicherheit gegen Datenverlust innerhalb eines Tages nicht ausreicht, wurde die Editor-DB alle 10 Minuten zusätzlich auf einen anderen Rechner repliziert. Bei dieser Replikation unter Lotus werden lediglich neue Dokumente kopiert, so dass die Datenübertragung sehr schnell erfolgt.

Inzwischen bestehen die Lotus-Server aus einem Cluster-System, so dass alle Datenbanken automatisch doppelt vorhanden sind und der Ausfall des gesamten Systems dadurch extrem unwahrscheinlich wird:



11. Datenschutz

Da es sich bei Bewerbungsdaten in der Regel um sensible Daten handelt, sind auch datenschutzrechtliche Aspekte von großer Bedeutung. Die für den Datenschutz eingesetzten Maßnahmen soll das folgende Schaubild verdeutlichen:



Selbstverständlich erfolgen alle Zugriffe auf die Datenbanken über das WWW ssl-verschlüsselt.

Die Eingaben zu den Bewerbungsunterlagen in die Input-DB sind nur als „Einlieferer“ möglich, d. h., die Daten können lediglich wie bei einer Urne abgelegt, aber hinterher nicht mehr eingesehen werden.

Alle Daten in der Datenbank RegCode können im Prinzip von jedem WWW-Browser gelesen werden. Da diese Datenbank jedoch keine Masken, das sind Anzeigevorlagen, enthält, können einem Browser auch keine Daten geliefert werden. Die Überprüfung des Registration-Codes erfolgt ausschließlich auf den Lotus-Servern.

Auf die Editor-DB wird zum einen von den Mitarbeiterinnen und Mitarbeitern in der Generalverwaltung über den Notes Client hochgradig verschlüsselt und zum anderen von den externen Gutachterinnen und Gutachtern über das WWW mit Passwort und ssl-verschlüsselt zugegriffen.

Die Zugriffssicherheit wird dabei von den Lotus-Servern gewährleistet.

12. Abschlussarbeiten

Die von den Gutachterinnen und Gutachtern ausgewählten Bewerberinnen und Bewerber wurden von den Mitarbeiterinnen und Mitarbeiter in der Generalverwaltung per E-Mail zu Vortragsveranstaltungen eingeladen. Die Absagen erfolgten automatisch per Programm (Agent) aus der Editor-DB.

Ende 2006 beginnt bereits die vierte Bewerbungskampagne für selbstständige Nachwuchsgruppen innerhalb der Max-Planck-Gesellschaft, so dass das elektronische Bewerbungssystem auch weiterhin erfolgreich eingesetzt wird.

13. GWDG und Lotus Software

Die GWDG setzt seit 1999 Lotus Software neben der bekannten Groupware-Funktionalität in zunehmendem Maß für organisatorische Aufgaben ein. Dazu gehören die folgenden Anwendungsbereiche:

- Raum- und Ressourcenbelegung
- Sitzungen, Protokolle, Wiedervorlagen
- wissenschaftliche Tagungen
- eBAR
- Online-Umfragen mit Auswertungen

- Datenschutzregister
- Erfassung und Abrechnung von Aufträgen
- Benutzerverwaltung
- Bewerbungen über das WWW

Die Einrichtung einer entsprechenden Lotus-Datenbank wird dabei von der GWDG allen Instituten als Dienstleistung angeboten.

Zielgenaue Verteilung, Konfiguration und Verwaltung von Software in heterogenen IT-Umgebungen

Bertram Smolny, Mario Gzuk, Tino Mai, Hartmut Arlt

Max-Planck-Institut für Biogeochemie, Jena

Zusammenfassung

Das aus der Open-Source-Community bekannte Projekt UA¹ wird um eine Inventarisierungsfunktion (sysinfo2) erweitert und so zu einem mächtigen Werkzeug bei der zielgenauen Verteilung, Konfiguration und Verwaltung von Software in heterogenen Umgebungen.

1. Motivation

»Life is really simple, but we insist on making it complicated.« (Konfuzius)

- Wer arbeitet mit welcher Software?
- Wer nimmt welche Lizenzen in Anspruch?
- Wer hat zuletzt Änderungen am Rechner vorgenommen?

1. <http://unattended.sourceforge.net/> [5]

Dieses sind einige der Fragen, die mit der Einführung der Kosten-Leistungs-Rechnung in der MPG relevant sind; Antworten darauf verursachen in den DV-Abteilungen eine Menge Arbeit, hat man keine Werkzeuge zur Hand, die einem das erleichtern können.

So verschieden Unternehmen und Institutionen sind, so divers sind auch ihre Anforderungen beim Verwalten von IT-Installationen. Für Unternehmen mit komplexeren IT-Infrastrukturen und einer größeren Anzahl von zu verwaltenden Computern sind konventionelle Installationen über eine CD nicht praktikabel.

Auch wächst die Gefahr, dass mit steigender Komplexität die Übersicht über die eigene Infrastruktur abhanden kommt. Dies kann insofern kritisch werden, da es einerseits Sicherheitslücken durch nicht oder nicht ausreichend gepatchte Systeme im eigenen Netz aufreißt, andererseits vor allem bei Verwendung kommerzieller Software der Gefahr des Lizenzmissbrauches möglich wird. Beides kann ernste wirtschaftliche Konsequenzen haben. Möchte man diesen Schwierigkeiten aus dem Weg gehen, so stellt sich die Aufgabe der zielgenauen Verteilung, Konfiguration und Verwaltung von Software in der eigenen IT-Infrastruktur.

2. Lebenszyklus einer IT-Installation

Nach Evard [1] ist der Lebenszyklus eines Computers stark bestimmt durch das Schwingen zwischen Neuinstallation und unbestimmtem Zustand. Um diesem Zustand entgegen zu wirken, werden vielerorts regelmäßig Neuinstallationen flächendeckend durchgeführt. Das kann mit Hilfe von Images erfolgen, von RIS², oder anderen Technologien, die möglichst Automatisierung bieten. Automatische Installation heißt aber auch im erweiterten Sinne, dass, um Fehler durch menschliche Schwächen zu vermeiden, alle notwendigen Entscheidungen im Vorfeld getroffen worden sind, dass auch bei unterschiedlichen Personen das Installationsergebnis vollkommen identisch ist, dass im Nachhinein keine Korrekturen per Hand mehr erforderlich sind und dass alle Arbeitsschritte unabhängig vom Bearbeiter protokolliert sind.

2. Remote Installation Service

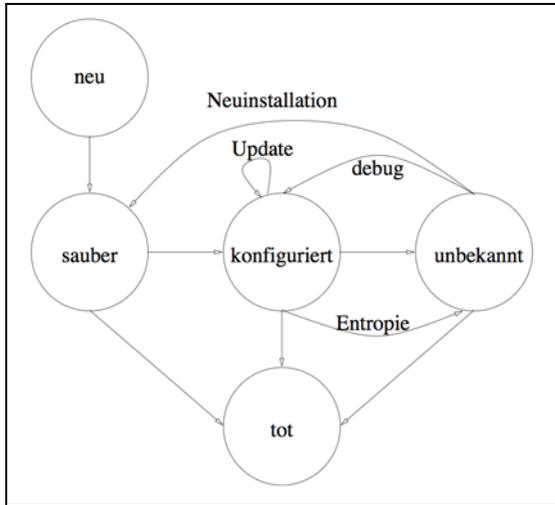


Abb. 1: Evard's Lebenszyklus eines Rechners

Am Markt sind derzeit einige Produkte zu finden, die bei der Lösung dieser Probleme in geeigneter Weise helfend zur Seite stehen. Allen gemein ist sehr häufig die ausschließliche Unterstützung einer einzigen Betriebssystemumgebung. So sind beispielsweise enteos NetInstall [2] und der Microsoft Systems Management Server in homogenen Umgebungen sehr leistungstark. Hat man nebenher aber auch, wie in Forschungsumgebungen üblich, Linux-Systeme im Einsatz, so bedarf es schnell eines weiteren Produktes, wie beispielsweise das Oracle Management Pack for Linux [4], welches jüngst ankündigt wurde. Als Konsequenz erhält man zu seiner heterogenen IT-Installation eine heterogene Administration, die für jede Betriebssystemumgebung in der Regel auch einen eigenen Workflow verlangt.

3. Disk-Imaging Deployment

Ein Deployment auf der Basis eines Disk-Images, einschließlich Betriebssystem, Anwendungen und Einstellungen, ist ein Weg, der sich schnell im Alleingang realisieren lässt. Im Trivialfall genügt ein `dd`³ einer Linux LiveCD. Der Markt hält aber auch hier ausreichend Produkte parat. Eine

3. `dd` erstellt 1:1-Kopien von Datenträgern. Egal ob Festplattenpartitionen, CDs oder DVDs, `dd` liest und schreibt blockweise. Da `dd` diese Blöcke nicht verarbeitet oder interpretiert, spielt es keine Rolle, um welches Dateisystem es sich handelt.

Installation liefert die Vorlage für weitere. Alle Installationen sind identisch und bieten identische Funktionalitäten auf allen Systemen. Ist man in der komfortablen Lage, eine homogene Hardwareausstattung zu besitzen, braucht es lediglich für jede Betriebssystemumgebung ein Image. Die Verwaltung beschränkt sich damit auf die Imagepflege. Eine solche homogene Hardwareausstattung mag zwar wünschenswert sein, dürfte jedoch in den seltensten Fällen der Realität entsprechen. In der Praxis bedeutet dies dann, für jeden Hardwaretyp ein Image vorzuhalten - vorzugsweise in verschiedenen (Treiber-)Versionen.

Nun wird die Verwaltung erheblich aufwändiger. Die wachsende Anzahl an Images muss einerseits gepflegt werden, andererseits muss immer das korrekte Image dem jeweiligen System zugeordnet werden. Die Übersicht und die eigene Motivation kann so schnell schwinden (siehe auch [9]).

4. Unattended Installation

Bei einer so genannten unattended Installation wird das komplette Setup eines Betriebssystems oder Programms durchlaufen, ohne dass währenddessen Interaktionen nötig sind. Angaben während der Installation, wie gewünschter Installationspfad, die Eingabe von Lizenzinformationen oder ähnliches entfallen, da diese Einstellungen zuvor in einem Skript, einer Antwortdatei oder in Kommando-Zeilenparametern festgelegt werden.

Die Installationszeit verringert sich in den meisten Fällen merklich, da eben nicht auf Eingaben gewartet werden muss und somit mehrere Installationen auch parallel ausgeführt werden können. Wird die gleiche Software auf mehreren Installationen benötigt, kann durch diese Automatisierung Zeit gespart werden, da diese zuvor nur einmal konfiguriert, dann aber beliebig oft und im Gegensatz zum Disk-Imaging auch auf unterschiedlichsten Systemen durchgeführt werden kann, ohne weitere Arbeitszeit zu beanspruchen. Die Fehlerrate sinkt drastisch.

5. Das Projekt »Unattended GUI«⁴

Inspiziert vom frei verfügbaren Projekt »Unattended, a WINDOWS deployment system« [5] wurde von Mario Gzuk auf Basis so genannter »unattended installations« im Jahr 2004 begonnen, ein System zu entwickeln, welches zunächst ein Deployment von MS Windows-Systemen in der heterogenen IT ermöglichen sollte. Bei diesem, zuerst als Freizeitprojekt gestar-

4. <http://de.wikipedia.org/wiki/GUI>

tetem Entwurf, ging es von Beginn an nicht nur um Installation, sondern auch um Management. Entstanden ist eine Middleware, die in ihrer heutigen Ausführung neben der Betriebssystem- und Softwareinstallation auch Verwaltungs- und Wartungsmöglichkeiten bietet. Neben der Installation von diversen Betriebssystemen wie Windows 2000/XP/XP-64bit/2003 sind auch automatische Installationen von verschiedenen Linux-Distributionen möglich. Mit einem stringenten Konzept der Modularität lässt sich die GUI in nahezu jede Umgebung einfügen. Das wird nicht nur durch die verschiedenen Authentifizierungsmethoden (LDAP, AD, Datenbank, IMAP, Radius etc.) erreicht, sondern auch durch die Unabhängigkeit von der verwendeten Datenbank (mysql, postgresql, mssql etc.). Nützliche Module wie z. B. die Verwaltung von DHCP-Einträgen im LDAP, runden die Flexibilität ab. Durch den modularen Aufbau und die Strukturierung einer einfachen API⁵ lässt sich auch mit geringem Aufwand ein eigenes Modul implementieren. Durch rechner-spezifische Module, so genannte Konnektoren, bietet sich eine gute Möglichkeit, die GUI mit bereits existierender Software im Unternehmen (z. B. Trouble-Ticket-Systemen wie Request Tracker [6]) zu verknüpfen.

Eine Einbindung der zentralen Switch-Verwaltung ermöglicht es z. B. auch, immer den aktuellen Standort des Rechners zu erfahren.

6. Das Projekt »Simo« und das Werkzeug »sysinfo2«

Das »System Inventory and Monitoring Tool Set«, kurz Simo [7], ist als eigenständiges Projekt im Jahre 2006 von Tino Mai und Hartmut Arlt am Max-Planck-Institut für Biogeochemie in Jena gestartet worden, mit dem Ziel, das Unattended GUI mit eigenständigen Tools zu erweitern bzw. bereits vorhandene Werkzeuge der community zur Verfügung zu stellen. Derzeitiger Entwicklungsschwerpunkt liegt auf dem Tool sysinfo2.

In Anlehnung an das psinfo-Projekt⁶ wurde am Max-Planck-Institut für Biogeochemie zuerst mit Skripten gearbeitet, die entsprechende Systemparameter einzelner Maschinen unter Windows auslesen und in Textdateien ablegen konnten. Aus Gründen der Flexibilität und Wartbarkeit wurde die Gedanke erwogen, doch ein Programm zu haben, das flexibel in den login-Pfad eingebunden werden kann und das die systembestimmenden Parameter in heterogenen Rechnerlandschaften so ausliest, dass die Ausgabe (am besten im

5. <http://de.wikipedia.org/wiki/API>

6. <http://www.sysinternals.com>

Format XML) auf ein Netzwerklaufwerk (Share) in einer definierbaren Form (z. B. Hostname.Modul.xml) schreibt.

Im Rahmen des »Unattended-Projektes« entstand das Kommandozeilen-Werkzeug »sysinfo2«, welches der automatischen Zusammenstellung von Systeminformationen dient. Die gewonnenen Informationen werden zum Zwecke einer datenbankgestützten Anwendung in maschinenlesbarer Form ausgegeben. Wahlweise kann dies in Plain Text (aus Kompatibilitätsgründen zur Vorgängerversion) oder in XML erfolgen. Das Ausgabeformat ist auf allen unterstützten Plattformen identisch gehalten. Bei der Entwicklung wurde darauf geachtet, zwischen plattformabhängigen und -unabhängigen Codeteilen zu trennen, und zwar auf allen Plattformen in exakt gleicher Art und Weise. Dies erleichtert nicht nur die Wartung, sondern auch die Erweiterbarkeit des Programms.

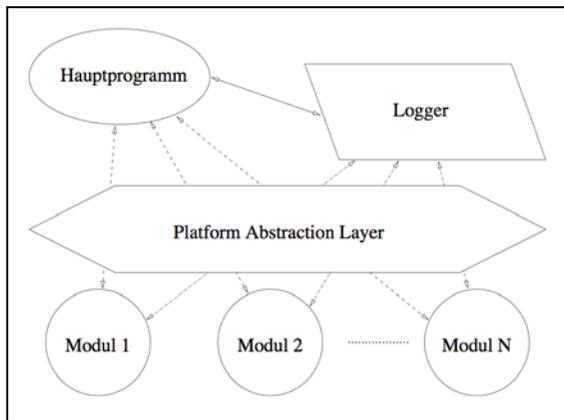


Abb. 2: Schematischer Aufbau von sysinfo2

Abbildung 2 zeigt in abstrakter Weise die Struktur des Programms mit Hauptprogramm, Logger, Platform Abstraction Layer (PAL) und Modulen. Die ersten beiden Instanzen sind plattformunabhängig implementiert, da ihre Funktionsweise auf allen Plattformen identisch sein soll. Die eigentliche Funktionalität des Programms schlummert in den Modulen, die aber plattformabhängig einbezogen verschiedenste Möglichkeiten des jeweiligen Systems nutzen, um die gewünschten Informationen zu erhalten. Da sie aber mit dem Hauptprogramm kommunizieren müssen und in einer einheitlichen Form vorliegen sollen, wurde ein Platform Abstraction Layer entworfen, welcher als Schnittstelle zwischen den beiden Sphären (plattformabhängig

<-> plattformunabhängig) fungiert. Das Hauptprogramm ist parametrisierbar, startet die Initialisierung der Module und steuert die verschiedenen Anfrageroutinen der Module. Anschließend teilt es dem Logger mit, dass mit der Ausgabe der Informationen begonnen werden kann. Der Logger kümmert sich nun selbstständig um die Ausgabe der Informationen. Dazu ruft er entsprechende Funktionen der Module auf, welche plattformunabhängig implementiert wurden, um eine einheitlich formatierte Ausgabe zu erzeugen.

Plattform	Betriebssystem / Architektur
Windows	ab Windows NT 5 (NT 4.0 mit Einschränkungen) x86; AMD64 und IA64
Linux	RPM-basierende Distributionen ab Kernel 2.6.x (Nicht RPM-basierende Distributionen sowie Kernel 2.4.x mit Einschränkungen); x86 und AMD64
Apple Macintosh	MacOS X ab Version 10.3 (Panther) PowerPC G4/G5 und Intel

Tab. 1: Unterstützte Plattformen

Tabelle 1 gibt einen Überblick über die derzeit unterstützten Plattformen und die damit verbundenen Mindestanforderungen bzw. Einschränkungen. Bei der Entwicklung musste eine Programmiersprache gewählt werden, die

- auf den Plattformen verfügbar ist,
- effizient ist und
- mit den Schnittstellen des Betriebssystems kommunizieren kann.

Abgeleitet aus dieser Anforderung fiel die Wahl auf C99⁷. Soweit dies der Compiler unterstützt, wurde die Konformität zu diesem Standard bei der Codegenerierung erzwungen. Diese Vorgehensweise resultiert aus der Anforderung, dass dieses Programm auf Produktionssystemem eingesetzt werden soll und somit ein Mindestmaß an Stabilität garantieren muss.

7. <http://de.wikipedia.org/wiki/C99>

Tabelle 2 zeigt die derzeit entwickelten Module und ihren Entwicklungsstand. Bis auf die Module Network und Display der Linux-Plattform sind alle Module vollständig implementiert und getestet worden.

Modul	Funktion	Entwicklungsstatus		
		Windows	Linux	OS X
OS	Betriebssysteminformation (Name, Version, Patchlevel)	fertig	fertig	fertig
CPU	Prozessorcharakteristika (Name, Typ, Frequenz)	fertig	fertig	fertig
Storage	Festplatteninformation (Name, Größe, Partitionierung)	fertig	fertig	fertig
Software	Installierte Software (Name, Version, Installationsdatum)	fertig	fertig	fertig
Network	Netzwerkkonfiguration (Netzwerkadapter, IP-Adresse, Subnetzmaske, Gateway, DHCP-Informationen)	fertig	Test	fertig
Display	Grafikkartencharakteristika (Name, Auflösung, Bildwiederholfrequenz)	fertig	Test	fertig
Memory	Arbeitsspeicherinformation physisch & virtuell (Größe, Ressourcen-Aufteilung)	fertig	fertig	fertig

Modul	Funktion	Entwicklungsstatus		
		Windows	Linux	OS X
Environ- ment	Arbeitsumgebungs- information (Variablennamen + dazugehörige Werte)	fertig	fertig	fertig

Tab. 2: Module von sysinfo2 und ihr Entwicklungsstand

7. Ausblick

- Wer hat zuletzt an diesem Rechner gearbeitet?
- Welche Fragen sind dazu im Trouble-Ticket-System offen?
- Mit welchem Port an welchem Switch im Hause ist der Rechner verbunden?

Besonders wertvoll werden die mit sysinfo2 und UA-GUI gewonnenen Informationen dann, wenn sie sich auch kombinieren lassen, und hier bieten sich weitere Anknüpfungspunkte für Entwicklungen, so dass das erreichte als Zwischentappe betrachtet werden kann. Weitere Partner sind aufgerufen, sich zu beteiligen.

In der Max-Planck-Gesellschaft setzen in der Zwischenzeit eine Reihe von Instituten von Hamburg bis Halle diese Software ein und beleben mit neuen Ideen deren Weiterentwicklung

So lassen sich mittlerweile mit UA-GUI auch Installations-CDs erstellen, um entfernte Rechner, die z. B. in Afrika unterwegs und nicht über das Netzwerk erreichbar sind, mit Installations-Paketen versorgen zu können. Dabei werden alle Abhängigkeiten für das Paket aufgelöst und eine ISO-Datei erstellt, von der sich einfach eine CD herstellen lässt, die auch den Postweg übersteht. Es wird eine Startskript angelegt, welches die Installationskripte in der richtigen Reihenfolge enthält und eine Autorun-Funktion aufruft.

8. Literatur

[1] Rémy Evard. An analysis of unix system configuration. 11th Systems Administration Conference (LISA 97), pages 179194, 1997

[2] enteo netinstall. <http://www.enteo.com/netinstall>

- [3] Microsoft systems management server. <http://www.microsoft.com/smsserver>
- [4] Oracle management pack for linux. http://www.oracle.com/technology/products/oem/omp_linux.html, 2007
- [5] Unattended, a windows deployment system. <http://unattended.sourceforge.net>
- [6] Unattended gui and managment system. <http://unattended-gui.sourceforge.net>
- [7] System inventory and monitoring tool set. <http://sourceforge.net/projects/simo>
- [8] Schritt-für-Schritt-Anleitung: <http://unattended.sourceforge.net/step-by-step.php>
- [9] „Desktop-Management bereitet CIOs Kopfzerbrechen“ (www.centratron.de)
http://www.cio.de/knowledgecenter/desktop_trends/832560/index.html

In der Reihe GWDG-Berichte sind zuletzt erschienen:

Nähere Informationen finden Sie im Internet unter

<http://www.gwdg.de/forschung/publikationen/gwdg-berichte>

- Nr. 40** *Plessner, Theo und Peter Wittenburg* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 1994
1995
- Nr. 41** *Brinkmeier, Fritz* (Hrsg.):
Rechner, Netze, Spezialisten. Vom Maschinenzentrum zum Kompetenzzentrum - Vorträge des Kolloquiums zum 25jährigen Bestehen der GWDG
1996
- Nr. 42** *Plessner, Theo und Peter Wittenburg* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 1995
1996
- Nr. 43** *Wall, Dieter* (Hrsg.):
Kostenrechnung im wissenschaftlichen Rechenzentrum - Das Göttinger Modell
1996
- Nr. 44** *Plessner, Theo und Peter Wittenburg* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 1996
1997
- Nr. 45** *Koke, Hartmut und Engelbert Ziegler* (Hrsg.):
13. DV-Treffen der Max-Planck-Institute - 21.-22. November 1996 in Göttingen
1997
- Nr. 46** **Jahresberichte 1994 bis 1996**
1997
- Nr. 47** *Heuer, Konrad, Eberhard Mönkeberg und Ulrich Schwardmann*:
Server-Betrieb mit Standard-PC-Hardware unter freien UNIX-Betriebssystemen
1998

- Nr. 48** *Haan, Oswald* (Hrsg.):
Göttinger Informatik Kolloquium - Vorträge aus den Jahren 1996/97
1998
- Nr. 49** *Koke, Hartmut und Engelbert Ziegler* (Hrsg.):
IT-Infrastruktur im wissenschaftlichen Umfeld - 14. DV-Treffen der Max-Planck-Institute, 20. - 21. November 1997 in Göttingen
1998
- Nr. 50** *Gerling, Rainer W.* (Hrsg.):
Datenschutz und neue Medien - Datenschutzzschulung am 25./26. Mai 1998
1998
- Nr. 51** *Plessner, Theo und Peter Wittenburg* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 1997
1998
- Nr. 52** *Heinzel, Stefan und Theo Plessner* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 1998
1999
- Nr. 53** *Kaspar, Friedbert und Hans-Ulrich Zimmermann* (Hrsg.):
Internet- und Intranet-Technologien in der wissenschaftlichen Datenverarbeitung - 15. DV-Treffen der Max-Planck-Institute, 18. - 20. November 1998 in Göttingen
1999
- Nr. 54** *Plessner, Theo und Helmut Hayd* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 1999
2000
- Nr. 55** *Kaspar, Friedbert und Hans-Ulrich Zimmermann* (Hrsg.):
Neue Technologien zur Nutzung von Netzdiensten - 16. DV-Treffen der Max-Planck-Institute, 17. - 19. November 1999 in Göttingen
2000

- Nr. 56** *Plessner, Theo und Helmut Hayd* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 2000
2001
- Nr. 57** *Hayd, Helmut und Rainer Kleinrensing* (Hrsg.):
17. und 18. DV-Treffen der Max-Planck-Institute
22. - 24. November 2000 in Göttingen
21. - 23. November 2001 in Göttingen
2002
- Nr. 58** *Plessner, Theo und Volker Macho* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 2001
2003
- Nr. 59** *Suchodoletz, Dirk von:*
Effizienter Betrieb großer Rechnerpools - Implementierung am Beispiel des Studierendennetzes an der Universität Göttingen
2003
- Nr. 60** *Haan, Oswald* (Hrsg.):
Erfahrungen mit den IBM-Parallelrechnersystemen RS/6000 SP und pSeries690
2003
- Nr. 61** *Rieger, Sebastian:*
Streaming-Media und Multicasting in drahtlosen Netzwerken - Untersuchung von Realisierungs- und Anwendungsmöglichkeiten
2003
- Nr. 62** *Kremer, Kurt und Volker Macho* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 2002
2003
- Nr. 63** *Kremer, Kurt und Volker Macho* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 2003
2004

- Nr. 64** *Koke, Hartmut* (Hrsg.):
GÖ* – Integriertes Informationsmanagement im heterogenen eScience-Umfeld: GÖ*-Vorantrag für die DFG-Förderinitiative „Leistungszentren für Forschungsinformation“
2004
- Nr. 65** *Koke, Hartmut* (Hrsg.):
GÖ* – Integriertes Informationsmanagement im heterogenen eScience-Umfeld: GÖ*-Hauptantrag für die DFG-Förderinitiative „Leistungszentren für Forschungsinformation“
2004
- Nr. 66** *Bussmann, Dietmar und Andreas Oberreuter* (Hrsg.):
19. und 20. DV-Treffen der Max-Planck-Institute
20. - 22. November 2002 in Göttingen
19. - 21. November 2003 in Göttingen
2004
- Nr. 67** *Gartmann, Christoph und Jochen Jähnke* (Hrsg.):
21. DV-Treffen der Max-Planck-Institute
17. - 19. November 2004 in Göttingen
2005
- Nr. 68** *Kremer, Kurt und Volker Macho* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 2004
2005
- Nr. 69** *Kremer, Kurt und Volker Macho* (Hrsg.):
Forschung und wissenschaftliches Rechnen - Beiträge zum Heinz-Billing-Preis 2005
2006
- Nr. 70** *Gartmann, Christoph und Jochen Jähnke* (Hrsg.):
22. DV-Treffen der Max-Planck-Institute
16. - 18. November 2005 in Göttingen
2006
- Nr. 71** *Hermann, Klaus und Jörg Kantel* (Hrsg.):
23. DV-Treffen der Max-Planck-Institute
15. - 17. November 2006 in Berlin
2007