

LDVZ – Nachrichten

Herausgeber:
Landesamt
für Datenverarbeitung und Statistik
Nordrhein-Westfalen

Redaktion:
Bianca Oswald,
Alfons Koegel

Kontakt:
Landesamt für Datenverarbeitung
und Statistik NRW
Postfach 10 11 05
40002 Düsseldorf,
Mauerstraße 51
40476 Düsseldorf

Telefon:
0211 9449-01
Telefax:
0211 442006
Internet:
<http://www.lds.nrw.de>
E-Mail:
poststelle@lds.nrw.de

Auflage:
1 200

© Landesamt
für Datenverarbeitung
und Statistik NRW,
Düsseldorf, 2005
Vervielfältigung und Verbreitung,
auch auszugsweise, mit Quellen-
angabe gestattet.

Bestell-Nr. Z 09 1 2005 51

ISSN 1616-377X

... in Kürze

**EIFEL – Eignungstest
für Feuerwehrleute
Ein Verfahren für das Institut
der Feuerwehr Nordrhein-Westfalen** **2**

**Oft verkannt – Der Praktikant
Ein Erfahrungsbericht eines
Praktikanten im LDS NRW** **3**

Schwerpunktt Themen

**Einsatz von
Open Source Software
im LDS NRW** **5**

**Web Service
– Technologie der Zukunft?** **10**

**Serverbereitstellung
in der Landesdaten-
verarbeitungszentrale** **14**

**Einrichtung sicherer
Verzeichnisse für Arbeits-
gruppen mit Windows-
Technologien** **18**

Mono: .Net für Linux **22**

**WAVE
Eine umfassende Unterstützung
für die Veranstaltungsplanung
und -durchführung** **25**

Das neue V-Modell XT **29**

**Agil und extrem (2)
Praktische Erfahrungen
mit agilen Methoden** **38**

Index 2000 – 2005 **48**

LDVZ-Nachrichten

... in Kürze

EIFEL – Eignungstest für Feuerwehrleute Ein Verfahren für das Institut der Feuerwehr Nordrhein-Westfalen

Für eine nutzbringende Teilnahme an den Seminaren des Instituts der Feuerwehr NRW (IdF) zur fachlichen Fort- und Weiterbildung sind typischerweise bestimmte Teilnahmevoraussetzungen zu erfüllen. Es erscheint daher sinnvoll, dass Interessenten, die an bestimmten Lehrgängen und Seminaren teilnehmen wollen, vorab testen können, ob sie die geforderten Voraussetzungen erfüllen. Dies soll im Internet auf komfortable Art und Weise geschehen.

Genau hier setzt EIFEL (Eignungstest für Feuerwehrleute) an: EIFEL ermöglicht einen datenbankgestützten Eignungstest zu bestimmten Lehrgängen bzw. Themen. Der Test liefert dabei eine Selbsteinschätzung für potenzielle Lehrgangsteilnehmer/-innen. Das Verfahren EIFEL unterstützt das IdF dabei, eine beliebige Anzahl möglicher Tests anzulegen und diese diversen Fachbereichen bzw. Themen zuzuordnen. Es wird die Möglichkeit geboten, eine beliebig große Anzahl von Fragen mit zugehörigen Antworten einzugeben. Es handelt sich hierbei um Fragen und Antworten nach dem Multiple-Choice Prinzip, wobei auch mehrere Antworten richtig sein können. Aus den im System erfassten Tests kann nun die/der potenzielle Lehrgangsteilnehmerin bzw. -teilnehmer den Test wählen, der für den von ihr/ihm gewünschten Lehrgang relevant ist.

Bei einem Selbsttest werden drei Phasen durchlaufen.

- In der ersten Phase wählt die/der interessierte Lehrgangsteilnehmer/-in einen Fachbereich und einen dazugehörigen Test. Nähere Informatio-

nen über den gewählten Test werden präsentiert.

- In der zweiten Phase wird der eigentliche Test durchgeführt und mögliche Fragen werden dem Interessenten nacheinander zur Beantwortung angezeigt.
- Die dritte Phase ist die Auswertung, hier wird der/dem möglichen Lehrgangsteilnehmer/-in angezeigt, wie viele Punkte im vorangehenden Test erreicht worden sind.

Da sich die/der potenzielle Teilnehmer/-in nun ein Bild darüber machen kann, ob sie/er die nötigen Voraussetzungen für den gewählten Lehrgang besitzt, liegt die Entscheidung bei ihr/ihm, sich für diesen Lehrgang anzumelden. Sollten die Voraussetzungen nicht erfüllt sein, besteht die Möglichkeit, diese durch einen vorgeschobenen Grundlagenlehrgang oder ein vergleichbares Seminar, zu erlangen.

EIFEL bietet zusammengefasst folgende Möglichkeiten:

- Erfassen einer beliebigen Zahl an Fachbereichen und Lehrgangstests
- Erfassen einer beliebigen Zahl an Fragen (inklusive den zugehörigen Antworten, gekennzeichnet als richtig bzw. falsch) zu einzelnen Lehrgangstests
- Pflege aller erfassten Daten
- zufällige Auswahl der zu stellenden Fragen aus dem gesamten Fragenkatalog zu einem Test
- Durchlaufen der Fragen und Antworten in zufälliger Reihenfolge
- Ergebnisausgabe (Bewertung) für den jeweiligen Test
- Ausgabe einer statistischen Auswertung
- Benutzerverwaltung (editieren der Tests sowie statistische Auswertungen nur für einen ausgewählten Personenkreis erlaubt)

Da das Verfahren in das Internetangebot des IdF eingebettet sein wird, wurde es auf der Basis einer MySQL-Da-

Lehrmittel - Selbsttest

Typprüfung - Brandklassen
Frage (1/2)
Welche Kombinationen aus Brandklasse und Beschreibung gibt es?

- Brandklasse A** - Brände fester Stoffe * hauptsächlich organischer Natur * verbrennen normalerweise unter Glutbildung (Glutbrände) * Holz, Papier, Stroh, Textilien, Kohle, Autoreifen, ...
- Brandklasse C** - Brände von flüssigen oder flüssig werdenden Stoffen (Flammbrände) * Benzin, Benzol, Öle, Fette, Lacke, Teer, Äther, Alkohol, Stearin, Parafin, ...
- Brandklasse D** - Brände von Metallen (Flammbrände) * Aluminium, Magnesium, Lithium, Natrium, Kalium und deren Legierungen, ...
- Brandklasse E** - Gasbrände (Flammbrände) * Methan, Propan, Wasserstoff, Acetylen, Erdgas, Stadtgas, ...

[weiter](#)

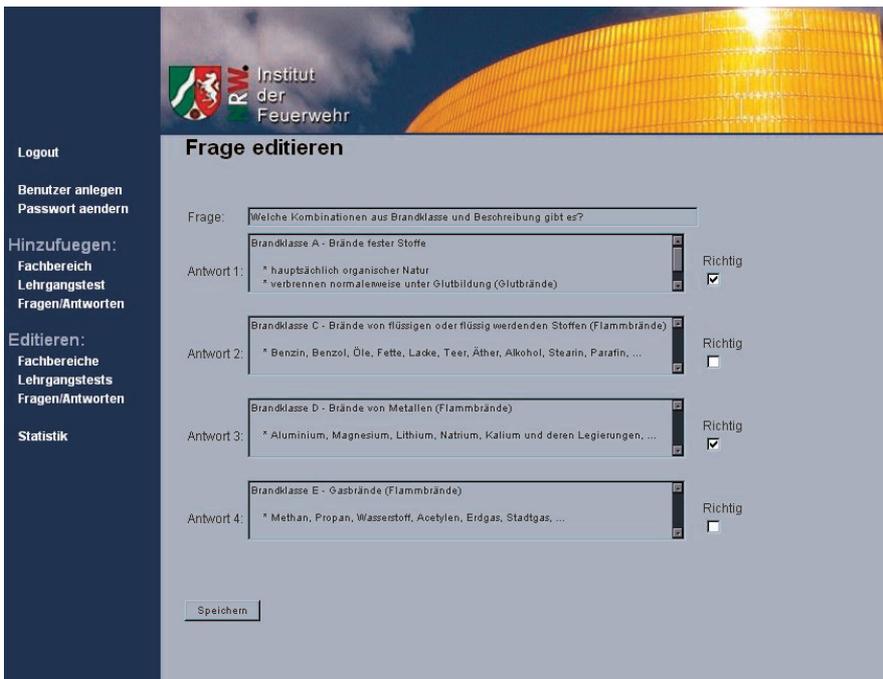
Lehrmittel - Selbsttest

Typprüfung - Brandklassen
Alle Fragen auf einen Blick

Welche Kombinationen aus Brandklasse und Beschreibung gibt es?
Ihre Antwort ist: *richtig*

Welches sind Beispiele für Flüssigkeiten oder flüssig werdende Stoffe der Brandklasse B (Flammbrände)?
Ihre Antwort ist: *falsch!*

Ergebnis: Peter Mustermann, Sie haben 1 von 2 Punkten.



tenbank entwickelt. Zur Darstellung der dynamischen Inhalte kommt die Scriptsprache PHP zum Einsatz. Diese ermöglicht eine zuverlässige und zugleich komfortable Anbindung an die zugrundeliegende Datenbank.

EIFEL wird, wie bereits angesprochen, für das Institut der Feuerwehr (IdF) entwickelt, das mit einem Personalbestand von etwa 115 Mitarbeiterinnen und Mitarbeitern die größte deutsche Feuerwehrausbildungseinrichtung ist. Bei einer Kapazität von ca. 310 Internatsplätzen werden hier die verschiedensten Lehrgänge und Seminare für berufliche und ehrenamtliche Angehörige der öffentlichen und privaten Feuerwehren angeboten und durchgeführt. Bei einem dermaßen umfangreichen Lehrgangsprogramm erscheint es unumgänglich, für vorgeschaltete Eignungstests ein möglichst automatisiertes Verfahren einzusetzen. Und es gibt auch bereits Überlegungen, das zunächst für den Selbsttest entwickelte EIFEL zu einem Verfahren für einen verbindlichen Eingangstest auszubauen.

Frank Staschok
Kontakt über die Mitautoren

Jürgen Sperling
Telefon: 0211 9449-6729
E-Mail: juergen.sperling@lds.nrw.de

Dipl.-Inform. Ulrich von Hagen
Telefon: 0211 9449-6706
E-Mail: ulrich.von-hagen@lds.nrw.de

Oft verkannt – Der Praktikant Ein Erfahrungsbericht eines Praktikanten im LDS NRW

Alles begann mit einer Initiativbewerbung per E-Mail beim LDS NRW. Kurz darauf erhielt ich auch schon einen Anruf vom zuständigen Personalsachbearbeiter, der sich meiner Bewerbung annahm. Er sprach mit mir über eventuelle Interessenschwerpunkte, welche die Auswahl des Bereiches beeinflussen sollten. Kurze Zeit später hatte ich dann auch ein Vorstellungsgespräch mit der zuständigen Person des heutigen Referats 253 – „Spezielle Anwendungen“ und mir wurde schnell klar, dass dies der Bereich meiner Wahl sein würde. Nachdem anschließend alle formalen und inhaltlichen Dinge besprochen waren, konnte mein Praktikum im September 2004 beginnen.

Das Projekt, welchem ich zugeteilt wurde, war EIFEL – Eignungstest für Feuerwehrleute (siehe hierzu auch Kurzbeitrag zum entsprechenden Verfahren in dieser Ausgabe der LDVZ-Nachrichten). Hier sollte eine Art Selbsttest für mögliche Lehrgangsteilnehmer/-innen des Instituts der Feuerwehr NRW (IdF) entwickelt werden. Ihnen sollte die Möglichkeit gegeben werden, die Teilnahmevoraussetzungen für bestimmte Lehrgänge anhand diverser Tests zu überprüfen. Die Entwicklung wurde über einen Zeitraum von vier Monaten angesetzt, was sich in etwa mit meiner Praktikumszeit deckte. Die Entwicklung dieses Verfahrens geschah auf der Grundlage einer MySQL-Datenbank mit Zuhilfenahme einer Fülle von PHP-Scripten. Die Datenmodellierung erfolgte mit Hilfe der Entity-Relationship-Methode. Die Entwicklung auf diesem Gebiet war neu für mich, programmierte ich doch bisher ausschließlich in Java und C/C++. Um eine solide Grundlage zu schaffen, wurde mir der Besuch mehrerer Lehrgänge ermöglicht. Als Entwicklungsumgebung wurde das ZEND-Studio gewählt, recht übersichtlich und mit allen nötigen Funktionen ausgestattet, erleichterte es die Entwicklung in vielerlei Hinsicht.

Während der Entwicklung bekam ich außerdem Einblick in Dinge, welche den Anwendungsbereitsstellungsstandard für die Verfahrensentwicklung in der LDVZ (ABS) betreffen. Ich fertigte Anwerferforderungen an, erstellte KM- und QS-Pläne, machte mir Gedanken zum Projekthandbuch, um schließlich alle nach diesem Standard geforderten Dokumente zu erhalten. Im späteren Verlauf erstellte ich für meine Programmiergruppe Programmierrichtlinien, die es ermöglichen sollten, den PHP-Code der jeweils anderen einfacher lesen zu können. Wert gelegt wurde hierbei auf die gleichartige Benennung eigener Variablen und Funktionen sowie auf die Art der Kommentierung. Zum Abschluss der Entwicklung musste das Verfahren noch getestet

LDVZ-Nachrichten

... in Kürze

werden. Hier kam ich erstmals in Kontakt mit Prüfspezifikationen und mit der Durchführung und Dokumentation verschiedener Tests. Anschließend galt es noch, das Verfahren in den Internet-auftritt des IdF einzubinden. Hierzu war eine Einarbeitung in Cascading Style Sheets nötig, was mir ebenfalls durch einen Lehrgang erleichtert wurde.

Während meines Praktikums lernte ich verschiedene agile Praktiken kennen. Angelehnt an die Vorgehensweise „Extreme Programming“ wurden Teile der Entwicklung durch das „Pair Programming“ – das gleichzeitige Arbeiten zweier Entwickler/-innen an einem Arbeitsplatz – unterstützt. Der Lerneffekt, der durch diese Art der Programmierung erzielt wird, ist beachtlich – gerade bei den hier vorliegenden unterschiedlichen Kenntnis- und Erfahrungsschwerpunkten. Fortlaufende Integration und teilweise fortlaufendes Testen waren weitere agile Prozesse, welche ich kennen lernen durfte.

Nun, was bringt einem Praktikanten die Wahl des LDS NRW als Praktikumsbetrieb?

Man hat hier die Möglichkeit mit netten Leuten im Team zu entwickeln, es werden Fortbildungen organisiert, man

wird gefördert und nicht ausschließlich gefordert. All dies klingt recht positiv und genau so werde ich es auch in Erinnerung behalten, wenn mein Praktikum abgeschlossen sein wird. Ich habe hier vieles gelernt, wie man bereits aus dem obigen Text entnehmen konnte und ich denke dass ist als Lohn auch ausreichend. Natürlich hätte ich es gerne gesehen, wenn auch noch eine gewisse Entlohnung in Form eines kleinen Gehalts vereinbar gewesen wäre, schließlich hockt man als Student nicht gerade auf einem Berg von Geld und die tägliche Anfahrt will auch bezahlt werden, aber in meinem Fall hat es auch gerade noch so geklappt. An dieser Stelle möchte ich auch gerne die Gelegenheit nutzen, meinen Kolleginnen und Kollegen und meinen Betriebsbetreuern im LDS NRW zu danken, es war eine gute Erfahrung und ich kann es auf jeden Fall weiterempfehlen.

Aber jetzt zu dem Punkt, was bringt ein Praktikant dem LDS NRW?

Hier sollte ich jetzt wohl besser meine Betriebsbetreuer zu Wort kommen lassen, doch ich versuche es mal mit eigenen Worten darzustellen. Ein Praktikant bringt frischen Wind in ein eingespieltes Entwicklerteam, wie jede(r)

andere neue Mitarbeiter/-in auch. Nicht ausschließlich durch seine Person, auch neue Ideen könnten durch seinen Beitrag den Weg ins Team finden, was positiv für beide Seiten wäre. Aus der Sicht eines Unternehmens, welches wirtschaftlich arbeiten sollte, stellt ein Praktikant, gerade wenn er von einer Hochschule oder Universität kommt, eine günstige und nutzbringende Arbeitskraft dar. Sein Wissen sollte in den Grundlagen soweit ausgebildet sein, dass er Aufgaben auch selbstständig ausführen kann und sich Mitarbeiter/-innen um eventuell wichtigere Dinge kümmern können.

Abschließend hier noch mal ein kurzer Dank: Das Praktikum hat mir viel gebracht, ich habe vieles gelernt und werde dafür in Zukunft auch mit Sicherheit genug Verwendung finden.

Ein Praktikant der Fontys technische Hogeschool in Venlo (NL).

Frank Staschok

Kontakt über:

Jürgen Sperling

Telefon: 0211 9449-6729

E-Mail: juergen.sperling@lds.nrw.de

Dipl.-Inform. Ulrich von Hagen

Telefon: 0211 9449-6706

E-Mail: ulrich.von-hagen@lds.nrw.de

Einsatz von Open Source Software im LDS NRW

Open Source Software ist seit einiger Zeit in den Medien ein heftig diskutiertes Thema, bei dem sich zwei oftmals zerstrittene „Lager“ gegenüberstehen: Zum einen die Befürworter rein proprietärer Software, zum anderen die Anhängerschaft von Open Source Software.

In vielen Rechenzentren sind allerdings schon seit geraumer Zeit Softwarekomponenten wie Linux oder Apache im Einsatz, ohne dass das Schlagwort „Open Source“ für die Entscheidung des Einsatzes entscheidend war. Technische und kaufmännische Gründe haben oftmals den Ausschlag gegeben.

Das LDS NRW betreibt seit jeher eine heterogene IT-Landschaft und versteht sich daher auch als Anbieter von Lösungen, die optimal auf die Kundenbedürfnisse abgestimmt sind – unabhängig davon, ob die Lösung auf rein proprietärer, rein offener Software oder auf einem Mischbetrieb beider „Welten“ aufsetzt. Somit ist auch Software aus dem Open Source-Umfeld ein essentieller Baustein im Produktangebot des größten Rechenzentrums der Landesverwaltung Nordrhein-Westfalen.

Im Folgenden wird der derzeitige Status einiger ausgewählter und etablierter Open Source-Produkte im LDS NRW dargestellt.

Vorbemerkungen

Open Source-Produkte haben einige vorteilhafte Eigenschaften, die gegenüber den Vorteilen von Alternativen zu bewerten sind. Zunächst ist hervorzuheben, dass für Open Source-Produkte keine Lizenzkosten anfallen und die Software beliebig oft installiert werden kann. Auch Updates und Upgrades können kostenlos bezogen werden.

Durch die Verfügbarkeit des Quelltextes entfällt die Abhängigkeit von nur einem Hersteller. Für Support und Weiterentwicklung können geeignete Dienstleister ausgewählt werden. Zusätzlich bietet die freie Verfügbarkeit auch die Möglichkeit die Software an die eigenen Anforderungen anzupassen und individuell zu erweitern.

Die Vielzahl der Entwickler sorgt für eine stetige Weiterentwicklung der Software und die schnelle Fehlerbeseitigung. Dies hat insbesondere in sicherheitskritischen Bereichen den Vorteil, dass auf diesem Weg Sicherheitslücken schnell geschlossen werden können. Da insbesondere Webserver

durch das Internet öffentlich zugängliche Systeme sind, ist hier der Sicherheitsaspekt besonders wichtig.

Serverbereitstellung unter Linux

Im Bereich der Serverbereitstellung kommen verschiedene Betriebssysteme und verschiedene Hardwareplattformen im LDS NRW zum Einsatz. Einen generellen Überblick über die Vorteile der einzelnen Systeme und Plattformen finden Sie im Beitrag „Serverbereitstellung in der Landesdatenverarbeitungszentrale“ in dieser Ausgabe der LDVZ-Nachrichten (s. S. xxff.). Das Betriebssystem Linux ist in den letzten Jahren ein wichtiger Baustein im Portfolio des LDS NRW geworden. Linux wird seit mehr als 5 Jahren im Serverbereich im LDS NRW eingesetzt.

Im Bereich der Fachverfahrensserver (Applikationsserver) und der Proxy/Firewall Installation des zentralen Internet-LVN-Übergangs ist Linux das derzeit am meisten eingesetzte Betriebssystem. Linux wird dabei vornehmlich als Enterprise Edition der beiden Distributionen RedHat und SuSE eingesetzt. Vorteile von Linux liegen unter anderem im modularen Aufbau, der eine relativ hohe Sicherheit der Systeme ermöglicht und ein auf die eingesetzte Applikationssoftware optimales abgestimmtes Umfeld anbietet, so dass Linux auch von vielen Anwendern als die bevorzugte Plattform für Applikationssoftware gesehen wird.

Als weitere Vorteile werden oftmals die Sicherheit der Systeme insbesondere in Bezug auf Viren genannt sowie die Hardwareunabhängigkeit. Im LDS NRW wird daher vornehmlich recht preisgünstige x86 Serverarchitektur verwendet. Erweitert wird das Angebot des LDS NRW durch eine Linux-Großrechnerimplementierung, die für spezielle Anwendungen in der Lage ist, eine höhere Verfügbarkeit zu gewährleisten. Insgesamt sind derzeit rund 100 Serversysteme unter Linux im LDS NRW im Betrieb. Schwerpunkt ist die IT-Produktion (Applikationsserver), die Proxy- und Firewalladministration sowie das Webhosting.

Die Wachstumsraten von Linuxsystemen im LDS NRW für den Teilbereich der IT-Produktion sind in dem folgenden Schaubild grob schematisch dargestellt. Hieraus ist ersichtlich, dass Linux das Betriebssystem im LDS NRW ist, das derzeit auch im LDS NRW mit Abstand am stärksten wächst und mittlerweile im Bereich der Fachverfahrensserver bereits Windows überholt hat (Stand: August 2004).

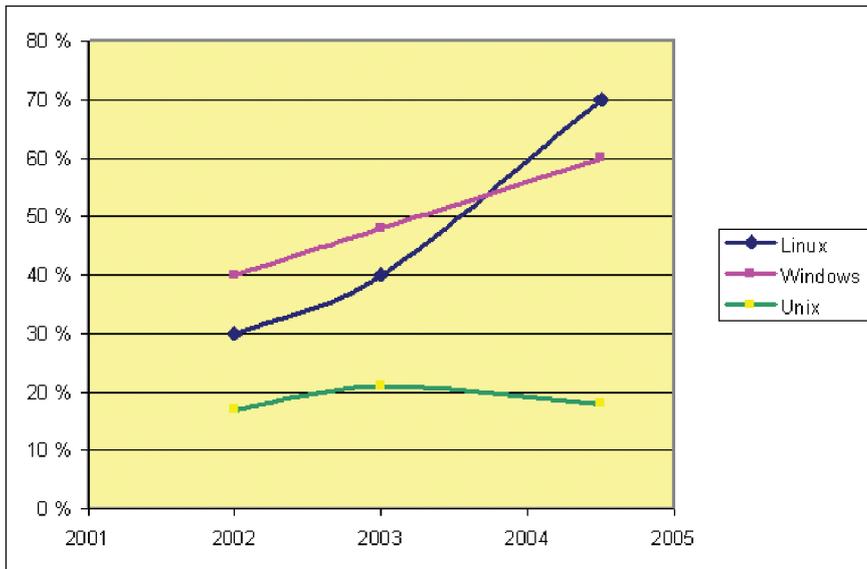


Abb. 1: Wachstumsraten von Linuxsystemen im LDS NRW

Auch die Art der betreuten Verfahren ist mittlerweile im Bereich der „unternehmenskritischen Anwendungen“ angekommen: Zentrale eGovernment Dienste (wie Mahnverfahren, virtuelle Poststelle u. a.), Content Management-systeme (CMS), Web-Portale für verschiedenste Ministerien und Behörden sowie Online-Statistik-Erfassungen und auch einige Datenbankserver setzen auf Linuxsystemen auf.

Einige Beispiele seien hier explizit genannt:

- Das Bildungsportal des MSJK, über das beispielsweise die zentrale Lehrereinstellung in NRW („LEO“, Lehrereinstellung Online) erfolgt. Hier werden mehrere leistungsfähige Linuxsysteme auf x86-Architektur mit je vier XEON-Prozessoren eingesetzt. Weiterhin werden Softwarekomponenten wie Apache, PHP und Tomcat verwendet (siehe unten).
- Das zentrale Content-Management für die Landesverwaltung: Im Rahmen des Betriebes des CMS für viele Kunden des LDS NRW werden mittlerweile über 10 leistungsfähige Server eingesetzt. Wenngleich die Software nps5 proprietär ist und die Datenhaltung zumindestens teilweise in einer Oracle-Datenbank erfolgt, setzt auch dieses strategische und wichtige Verfahren auf Linux-Be-

triebssystemen sowie Apache und Tomcat zur Darstellung der einzelnen Web-Präsentationen auf. Somit ist dieses CMS ein gutes Beispiel für das Zusammenspiel von freier und proprietärer Software.

- Genesis-Online, Statistik-Online und Justizstatistik-Online: Zentrale Statistikverfahren werden mittlerweile Online im LDS NRW angeboten: Hier wird in aller Regel der Web-Anteil über ein Linux-System mit Apache, PHP, Tomcat und z. T. Java-Komponenten realisiert. Die

Datenhaltung erfolgt z. T. in proprietären Datenbankformaten wie ADABAS oder aber mittels MySQL.

Generell ergibt sich somit für Fachverfahrensserver, die auf Webtechnik aufsetzen, und auch für eGovernmentverfahren ein typisches Bild für die logische Struktur, wie es in Abbildung 2 dargestellt ist (s. u.).

Software für Web-Anwendungen

Ein sicher ebenso bekanntes Open Source-Projekt wie Linux ist der Apache-Webserver. Release 1.0 wurde 1995 veröffentlicht und wird seitdem stetig weiterentwickelt. Er bietet eine Vielzahl an Konfigurationsmöglichkeiten und zeichnet sich durch gute Performance aus. Weiterhin existiert eine große Bandbreite an Modulen für die verschiedensten Zwecke (PHP, PERL, SSL...). Der Apache ist weltweit der verbreitetste Webserver und besitzt einen Anteil von fast 70 % wie die Netcraft-Statistik zeigt. Im LDS NRW liegt der Anteil von rund 800 Webangeboten sogar über 90 %.

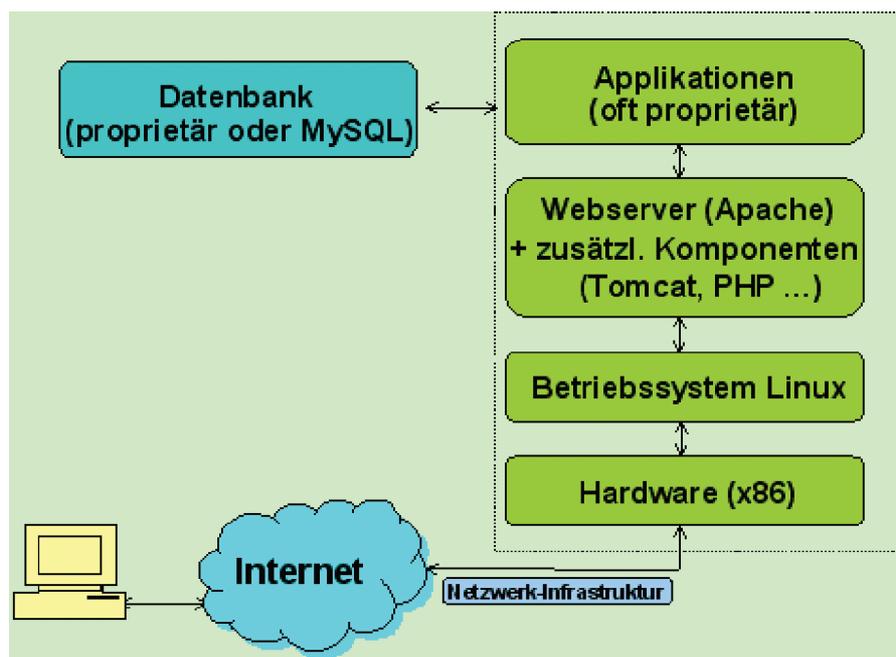


Abb. 2: Logische Struktur

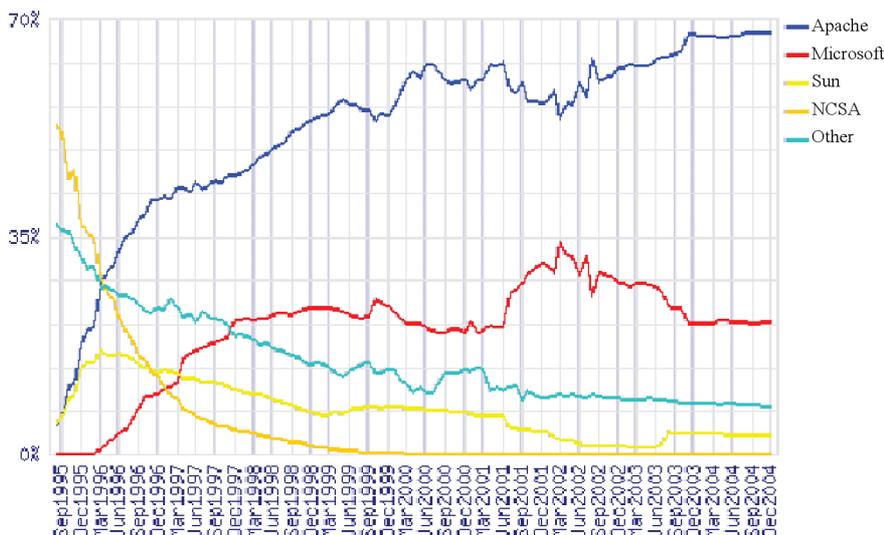


Abb. 3: Quelle: <http://news.netcraft.com/archives/2004/12/index.html>

Wird der Apache-Webserver unter Linux in Kombination mit den Softwareprodukten MySQL und PHP eingesetzt, bezeichnet man diesen Zusammenschluss als LAMP, einem Akronym für Linux, Apache, MySQL und PHP.

MySQL ist eine bekannte und gebräuchliche Open Source-Datenbank, die 1996 der Öffentlichkeit vorgestellt wurde. Seitdem erfreut sie sich bei Linux-Anwendern sowie Webdesignern größter Beliebtheit. Auch die Features des Datenbanksystems werden stetig weiterentwickelt. Unter den Kunden des LDS NRW werden innerhalb der Webangebote häufig MySQL-Datenbanken eingesetzt.

Da seit einigen Jahren kaum noch statische Webseiten präsentiert werden, sondern ein stetiger Wechsel zu dynamischen und interaktiven Webanwendungen vollzogen wurde, besonders auch im Zusammenhang mit eGovernment-Anwendungen, gewinnen die Open Source-Programmiersprachen immer mehr an Bedeutung:

PHP ist eine populäre Open Source-Skriptsprache speziell für Webentwicklungen. PHP lässt sich in HTML einbinden und ein wichtiger Aspekt dieser Sprache ist es, Webentwicklern die Möglichkeit zu geben, schnell dynamisch generierte Webseiten zu erzeugen. Eine große Stärke von PHP ist die Unterstützung von Datenbankabfragen

und es bietet sich so die Möglichkeit der einfachen Erstellung von datenbankgestützten Webseiten.

Die erste Version der Programmiersprache **Perl** (Practical Extraction and Report Language) wurde 1987 durch Larry Wall veröffentlicht. Hauptziel war es, mit Perl eine Skriptsprache zu entwickeln, die es ermöglichen sollte, Textdateien komfortabel zu bearbeiten. Im Laufe der Zeit wurde das CPAN (Comprehensive Perl Archive Network) geschaffen, ein Onlineverzeichnis für Perl-Quellcode, Dokumentationen, Skripte sowie Module und Erweiterungen. Hier wird die Kooperation innerhalb der Open Source-Community besonders deutlich. Mittlerweile ist Perl zu einer der wichtigsten Programmiersprachen für Systemadministration und Webprogrammierung geworden.

Um mittels serverseitiger Java-Programme dynamische Webangebote und Webapplikationen realisieren zu können, werden sowohl Java Servlets als auch JavaServer Pages (JSP) eingesetzt. Bei beiden handelt es sich um Java-Komponenten, die von einem Java-basierten Webserver unterstützt werden müssen. Dazu wird im LDS NRW die Software **Tomcat** auf einigen Webservern eingesetzt. Tomcat ist ein Open Source-Container für Java-basierte Web-Anwendungen, der Servlet- und JSP-Web-Anwendungen ausführt.

Er wird als offizielle Referenzimplementierung für die Servlet- und JSP-Spezifikationen von der Apache Software Foundation unter dem Projektnamen "Jakarta" entwickelt.

Exemplarisch seien an dieser Stelle noch einige Webanwendungen aufgeführt, die als Open Source Software vorliegen und auch auf den Webservern des LDS NRW eingesetzt werden. Für die schwarzen Bretter und Foren des LDS NRW, die täglich mehr als 20 000 Zugriffe verzeichnen, wird die Software **phpBB** eingesetzt. Diese basiert wiederum auf den Open Source-Produkten PHP und MySQL. Die phpBB-Community stellt laufend neue Templates und Module bereit, mit denen die phpBB-Foren individualisiert werden können. Auch die Vielzahl an Features, die diese Anwendung bietet, ist bemerkenswert.

Für die Auswertung von Webserver-Logfiles zum Erstellen von Webstatistiken sind auf den LDS NRW-Webservern die Open Source Tools **Webalizer** und **Analog** im Einsatz: Der Webalizer ist ein Programm zur Darstellung von Zugriffsstatistiken einer Webseite. Die Auswertungsmöglichkeiten sind recht umfangreich und lassen sich über die Konfigurationsdatei des Webalizers und des Webserver einrichten. Auch die Software **Analog** ermöglicht die Aufbereitung und Darstellung von Zugriffsstatistiken. Auch hier lassen sich die Reports und Grafiken umfassend konfigurieren.

Der auf einem Webserver des LDS NRW eingesetzte **MapServer** ist ein Tool, das die Möglichkeit der Navigation in Karten bietet. Er verarbeitet etliche der in diesem Bereich verwendeten Standardformate (GIS, ESRI) zur Beschreibung von Karten. Auch die Ausgabe lässt sich in Standardformaten konfigurieren (TIFF/GeoTIFF, GIF, PNG, ERDAS, JPEG). Die Grundlage dieses Systems ist der University of Minnesota MapServer, eine Entwicklung der – wie der Name schon sagt – University of Minnesota.

Weitere Produkte wie beispielsweise Shopsysteme, Auktionsplattformen und Webchats werden regelmäßig gesichtet und getestet. Sie können daher bei Bedarf den Kunden ohne Schwierigkeit zur Verfügung gestellt werden.

E-Mail und Spam-Bekämpfung

Gerade im Bereich E-Mail sind Open Source Programme seit den Anfängen des Internets stark vertreten. Der meistgenutzte Mail Transfer Agent (MTA), ein für die Übertragung von E-Mails verantwortliches Programm, ist immer noch das Open Source-Produkt sendmail, das seit 1983 entwickelt wird und mittlerweile in der Version 8 verfügbar ist. Sendmail ist ein sehr flexibles, damit aber auch kompliziert zu betreibendes Programm. Neuere MTAs wie exim oder postfix – ebenfalls Open Source Programme – ersetzen daher neuerdings sendmail an vielen Stellen. Das LDS NRW setzt, gerade wegen der hohen Flexibilität, für den Betrieb der zentralen E-Mail-Struktur des Landesverwaltungsnetzes und für den Übergang ins Internet auf sendmail. Dieser zentrale E-Mail Backbone (Abbildung 4) basiert im Wesentlichen auf Linux-Rechnern und verbindet ca. 500 Landesbehörden untereinander. Pro Monat verteilt diese Struktur in etwa 5 000 000 E-Mails.

Neben der eigentlichen E-Mail-Verteilung und der leider notwendigen Überprüfung auf Viren kann diese Struktur aber auch zum Schutz vor einem weiteren Übel im heutigen E-Mail-Verkehr eingesetzt werden: Zum Schutz vor Spam. Spam bezeichnet massenhaft unaufgefordert zugesandte E-Mails, die Postfächer verstopfen und die Empfänger eigentlich nur nerven (siehe auch

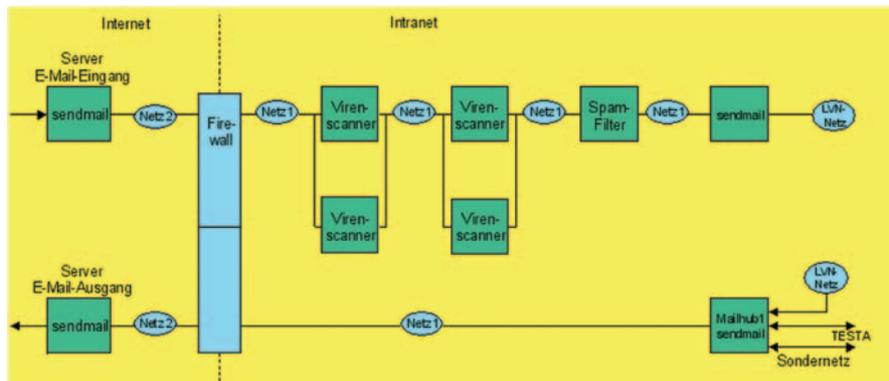


Abb. 4: Zentraler E-Mail Backbone

„SPAM – Wenn der Postmann zwei(tausend)mal klingelt“ aus den LDVZ-Nachrichten 2/2003). Teil des E-Mail Backbones aus Abbildung 4 ist ein Spam-Filter, der von jeder Landesbehörde in NRW optional genutzt werden kann. Der Spam-Filter analysiert die E-Mails und markiert Spam so, dass ein Empfänger mit Hilfe seines E-Mail-Programms diese problemlos herausfiltern und ignorieren kann. Der vom LDS NRW eingesetzte Spam-Filter basiert auf einem weiteren Open Source Programm: Dem SpamAssassin.

Spam-Filter

Der SpamAssassin ist ein unter der Apache Software License veröffentlichtes Open Source Programm zur Spam-Erkennung. Dazu werden E-Mails anhand eines Regelsatzes untersucht und auf Spam-Wahrscheinlichkeit bewertet. Ein großer und bewährter Regelsatz wird dabei mit dem Programm mitgeliefert. Die Stärke von SpamAssassin ist jedoch die Erweiterbarkeit dieses Regelsatzes, die es erlaubt, das Wissen der Administratoren im LDS NRW zur Spam-Erkennung zu nutzen und somit den Spam-Schutz entscheidend zu verbessern.

Die Spam-Analyse berechnet zu einer E-Mail eine Spam-Kennzahl, anhand derer eine Zuordnung Spam oder kein Spam geschehen kann. Dazu wird jede Regel des Regelsatzes auf die E-Mail angewendet. Trifft die Regel zu, wird die Kennzahl um einen Wert, der der Regel zugeordnet ist, verändert. So überprüft beispielsweise die Standardregel NUMERIC_HTTP_ADDR, ob in der E-Mail auf eine Internetadresse verwiesen wird, die nur numerisch (z. B. 193.159.219.135 anstelle von www.nrw.de) angegeben ist. Trifft diese Regel zu, wird die Kennzahl um 1,565 Punkte angehoben. Überschreitet die Kennzahl den Wert 5, so wird die E-Mail als Spam angesehen und in dem E-Mail-Header als solche markiert:

```
X-Spam-Status: Yes, hits=7.5 required=5.0 tests = BAYES_99, BIZ_TLD, FORGED_YAHOO_RCVD, HTML_50_60, HTML_MESSAGE, MIME_TML_ONLY
```

Diese Zeile gibt z. B. an, dass ein Wert von 7,5 erreicht wurde und die E-Mail damit als Spam angesehen wird. Zusätzlich ist angegeben, welche Regeln erfolgreich waren. Die Detailanalyse dieser E-Mail zeigt unten stehende tabellarische Darstellung:

Punkte	Regel	Beschreibung
0.1	HTML_MESSAGE	BODY: Nachricht enthält HTML
5.4	BAYES_99	BODY: Spamwahrscheinlichkeit nach Bayes-Test: 99-100% [score: 1.0000]
0.3	MIME_HTML_ONLY	BODY: MIME-Nachricht besteht nur aus HTML
0.1	HTML_50_60	BODY: Nachricht enthält zwischen 50 und 60% HTML
0.7	BIZ_TLD	URI: Hyperlink mit ".biz" am Ende
0.9	FORGED_YAHOO_RCVD	Gefälschte "Received"-Kopfzeile von yahoo.com gefunden

Eine Besonderheit dabei ist die Regel BAYES_99. Der hier eingesetzte Bayes-Test des SpamAssassin kann aus E-Mails lernen, die eindeutig Spam sind und somit selbstständig Spam erkennen. Dieser Lernschritt geschieht entweder automatisch, wenn der Regelsatz eine E-Mail mit sehr hoher Wahrscheinlichkeit als Spam erkennt, oder kann „per Hand“ angestoßen werden. Eine E-Mail mit Spamwahrscheinlichkeit 0 bekommt von dem Bayes-Filter einen Wert von -4,9 zugewiesen. In dem obigen Beispiel liegt die Wahrscheinlichkeit zwischen 99 % und 100 %, so dass ein Wert von 5,4 genutzt wird.

Die Zuordnung, ob eine E-Mail Spam ist oder nicht, ist nicht einfach festzustellen. Spam-Versender kennen die eingesetzten Spam-Filter und damit auch den SpamAssassin und passen die Spam-Mails an diese Programme an. Zudem muss bei der Analyse vermieden werden, dass E-Mails als Spam erkannt werden, die kein Spam sind. Hier jedoch hilft die offene Architektur des Open Source Programms SpamAssassin. Zum einen können die Administratoren den Bayes-Filter mit neu entdeckten Spam-Mails füttern und so die Erkennung verbessern. Dies alleine ist jedoch nicht ausreichend, da vor dem Erkennen einer solchen E-Mail diese

Regeln können auf die Besonderheiten der Landesverwaltung oder einzelner Behörden eingehen und somit mit einem kleinen, aber entscheidenden Beitrag zur Spam-Analyse beitragen. Eine solche Regel kann z. B. so aussehen:

```
uri SUSP_TLD /^(?:https?:\w|mailto:)[^\v]+\.(ms|pn|vu|vg)(?:\v|$)/iscore
SUSP_TLD 2.5describe SUSP_TLD
Verdächtige TopLevelDomain in Link
```

Diese Regel besagt, dass E-Mails, die Verweise auf Internet-Adressen, die auf .ms, .pn, .vu oder .vg enden, enthalten (z. B.: <http://www.ruboff.ouch.ms/>), verdächtig sind und somit die Kennzahl um 2,5 Punkte angehoben wird. So eine Regel ist möglich, da davon auszugehen ist, dass die Landesverwaltung wenig Kontakt zu z. B. Montserrat, einer Insel in der Karibischen See mit dem Länderkürzel .ms, hat. Wichtig ist jedoch, dass diese Regel nicht automatisch eine E-Mail, auf die die Regel zutrifft, als Spam behandelt. Nur wenn auch andere Regeln erfüllt sind, wird die Punktzahl von 5 erreicht, ab der die E-Mail als Spam angesehen wird.

Diese regelmäßig vorgenommenen Anpassungen der Spam-Analyse zahlen sich aus. Abbildung 5 zeigt eine Übersicht über die Spamerkennung an einem typischen Werktag.

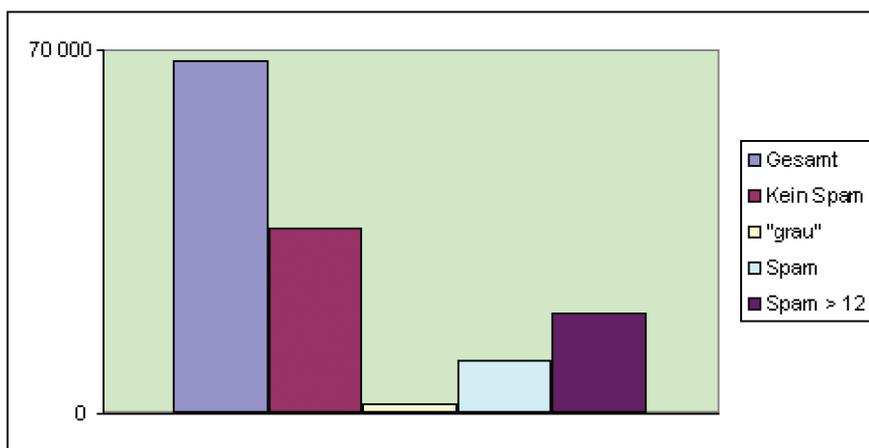


Abb. 5: Spamerkennung an einem typischen Werktag

schon an viele Empfänger im Land verschickt worden sein konnte. Zum andern besteht aber auch die Möglichkeit, durch eigene Regeln die Spam-Erkennung entscheidend zu verbessern. Diese

Von den etwa 70 000 überprüften E-Mails sind etwas mehr als die Hälfte eindeutig kein Spam (Kennzahl < 0). Fast 30 000 E-Mails sind eindeutig Spam (Kennzahl > 5), davon sind fast

20 000 E-Mails so eindeutig Spam, dass sie eine Kennzahl von über 12 zugeordnet bekommen. Die Grauzone – die E-Mails mit einer Kennzahl zwischen 0 und 5 – besteht aus weniger als 2 000 E-Mails. Gerade diese Zahl zeigt, wie erfolgreich SpamAssassin durch die offene, anpassbare Architektur ist, die typisch für Open Source Software ist.

Von den etwa 500 Landesbehörden, die einen E-Mail-Zugang über das LDS NRW haben, nutzen schon 388 den LDS-NRW-Spam-Filter und profitieren somit von Open Source. Informationen, wie auch Ihre Behörde den Spam-Filter nutzen kann, erhalten Sie bei dem untenstehenden Ansprechpartner.

Weitergehende Informationen

Das LDS NRW beteiligt sich aktiv an der Open Source-Initiative der Landesverwaltung NRW. Informationen sind unter <http://lv.linux.nrw.de/> zu finden.

Das LDS NRW bietet im Rahmen des IT- Fortbildungsprogramms des Innenministeriums NRW regelmäßig eine zweitägige Informationsveranstaltung rund um das Thema „Einsatz von Open Source Software“ sowie einen einwöchigen Administratorenkurs zum Thema „Linux-Systemverwaltung“ an. Weitere Informationen dazu können unter <http://www.it-prog.lds.nrw.de/> abgerufen werden.

Linux

Dr. Frank Dillmann
 Telefon: 0211 9449-2680
 E-Mail: frank.dillmann@lds.nrw.nrw.de

Web

Susanne Schmitz
 Telefon: 0211 9449-2060
 E-Mail: susanne.schmitz@lds.nrw.nrw.de

Mail

Dr. Thomas List
 Telefon: 0211 9449-3438
 E-Mail: thomas.list@lds.nrw.nrw.de

Web Service

– Technologie der Zukunft?

Seit einiger Zeit ist der Begriff der Web Services in aller Munde. Hohe Erwartungen werden an diese neue Technologie geknüpft: Um die Lösung aller Integrationsprobleme soll es sich handeln, kostengünstig, einfach, schnell zu implementieren, Investitionsschutz bestehender Anwendungen bieten, standardisiert, wiederverwendbar und plattformunabhängig sein, um nur einige zu nennen. Kurz: Web Services werden als die Technologie gehandelt, auf die die DV-Welt schon so lange gewartet hat.

Wirft man jedoch einen Blick in die Praxis stellt man fest, dass es gewisse begriffliche Verwirrungen darüber gibt, was unter einem Web Service zu verstehen ist. So zeigt eine Gartner-Untersuchung, dass viele Firmen schon die Nutzung von XML zum Datenaustausch als „Web Service“ bezeichnen. Auch gibt es bisher sehr wenig „echte“ Web Service Implementierungen, was angesichts der o. g. „Web Service Euphorie“ zunächst überrascht.

Ausgehend von einer Beschreibung der Web Service-Technologie soll dieser Artikel beleuchten, welche Probleme Web Services heute lösen können und wo Defizite bestehen.

Web Services als standardisierte Integrationstechnologie

Die Idee

Heute stehen wir vor der Situation, dass mehr oder weniger jede Aufgabe einer Verwaltung oder eines Unternehmens in irgendeiner Form DV-gestützt abgewickelt wird. Zur Optimierung von Geschäftsprozessen innerhalb einer Verwaltung/eines Unternehmens oder/und zwischen Verwaltungen/Unternehmen und ihren Partnern reicht dies aber nicht aus. Eine solche Aufgabe kann nur durch anwendungsübergreifende Integration bewältigt werden.

Folgerichtig wurden in der Vergangenheit unter den Stichworten Application-to-Application-Integration (A2A), Enterprise Application Integration (EAI), Business-to-Business-Integration (B2B) u. Ä. verschiedene Integrationsansätze für Softwaresysteme kreiert. Dabei handelt es sich sowohl um verschiedene Produkte diverser Hersteller als auch individuell zugeschnittene Lösungsansätze.

So nützlich diese Ansätze auf den ersten Blick sein mögen, so problematisch sind sie in ihrer Umsetzung: Die am Markt angebotenen Produkte sind nicht standardisiert und zudem teuer und schwer zu implementieren. Die individuell zugeschnittenen Lösungsansätze sind dagegen kaum änderbar, da

aufgrund der Abhängigkeiten bei einem solchen Vorhaben zu viele Seiteneffekte zu beachten sind. All dies ist mit den Anforderungen an moderne Verwaltungs- und Unternehmenssteuerung nur schwer zu vereinbaren.

Erforderlich ist vielmehr eine serviceorientierte Integrationsarchitektur, in der sich einzelne Komponenten unabhängig von anderen ändern können und die auf Standards basiert, an die sich alle großen Anbieter halten. Einen solchen Ansatz sollen Web Services liefern.

Was versteht man nun unter einem Web Service?

Definitionen und Standards

Eine allgemein gültige Definition gibt es nicht, wohl aber eine Reihe von Einzeldefinitionen. So definiert beispielsweise das W3C-Konsortium:

„A Web service is a software system identified by a URI, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using XML based messages conveyed by Internet protocols.“¹⁾

und spricht hiermit bereits die zentrale Web Service-Technologie XML an.

Die **eXtended Markup Language (XML)** ist ein Standard zur Definition von Formaten für den Datenaustausch, der die Möglichkeit bietet, auch komplexe Datenstrukturen und Abhängigkeiten zwischen Datensätzen innerhalb eines Dokuments abzubilden. Dabei wird die Struktur der XML-Daten in so genannten XML-Schema festgehalten. Die Ablage erfolgt jeweils als Textdatei, die nicht nur maschinen- sondern prinzipiell auch durch nicht DV-Spezialisten lesbar ist und mittels einfacher Texteditoren bearbeitet werden kann.

Verfolgt man das Ziel Anwendungen bzw. Komponenten über Systemplattformen hinweg zu integrieren, reicht die XML-Spezifikation selber nicht aus. Für ein solches Vorhaben ist es vielmehr notwendig festzulegen, auf welche Weise die Informationen im Rahmen der Integration transportiert werden, wie die konkreten Schnittstellen aussehen und,

1) Quelle: www.w3.org: Web Services Architecture Requirements, 1.1: What is a Web Service?

sofern man eine breite Nutzung erreichen möchte, auch wo die Beschreibungen aufzufinden sind. Dies führt zu den XML-basierten Web Service-Standards SOAP, WSDL und UDDI, die im Folgenden beschrieben werden.

Um im Rahmen der angestrebten Integration Informationen von einer Quelle zum Ziel überhaupt transportieren zu können, wurde das Nachrichtenaustauschformat **Simple Object Access Protokoll (SOAP)** entwickelt. Eine SOAP-Nachricht besteht grob aus einem Datenbereich (SOAP Body), der den eigentlichen Inhalt der Nachricht, also die Operationen und Parameter, verschlüsselt. Darüber hinaus können Informationen über Sicherheit, Transaktionen etc. in einem Kopf (SOAP Header) abgelegt werden. Beide Bereiche werden in einem Umschlag (SOAP Envelope), der zusätzliche Verwaltungsinformationen enthält, zusammengefasst. Wichtig ist, dass mit SOAP kein eigenes Transportprotokoll „erfunden“ wurde, sondern mit der sog. Bindung auf existierende Protokolle, wie z. B. dem HyperText Transfer Protocol (http), aufgesetzt wird.

Über das reine Nachrichtenaustauschformat hinaus, benötigt jede Art von Integration ein **Application Programmer Interface (API)**, das die konkrete Schnittstelle mit all seinen spezifischen Operationen und Eingabe-/Ausgabewerten beschreibt. Im Fall von Web Services existiert hierzu ein ebenfalls XML-basierter Standard, die **Web Service Description Language (WSDL)**. Über die eigentliche Schnittstelle hinaus wird mittels WSDL auch informiert, wie der Zugriff auf den Dienst möglich ist, welche Protokolle genutzt werden etc. Dabei wird WSDL in der Regel nicht „zu Fuß“ erzeugt, sondern entweder mittels Tools aus existierendem Code oder über grafische Editoren generiert. Da WSDL an SOAP gebunden ist, kann der SOAP-Kommunikationscode für eine konkrete Programmiersprache ebenfalls automatisch aus einer WSDL-Datei generiert werden.

Steht ein solcher Web Service zur Verfügung, stellt sich als nächstes die Frage, wie Informationen über seine Existenz und Funktionsweise einem breiten Nutzerkreis zugänglich gemacht werden. Die Lösung innerhalb der Web Services Architektur heißt **Universal Description, Discovery and Integration (UDDI)**. Mit den Möglichkeiten für die Registrierung und Suche wird UDDI auch als die „gelben Seiten“ für Web Services bezeichnet.

Neben den beschriebenen Web Service – Basisstandards gibt es verschiedene Erweiterungen, die sich auf Einzelaspekte komplexer Integrationen beziehen. Sie beschäftigen sich genauer mit Fragen der Sicherheit, der Behandlung von Transaktionen, der Workflow-Modellierung sowie der zentralen Überwachung und Verwaltung verteilter Web Service-Anwendungen. Die zugehörigen Standards befinden sich teilweise noch in der Entwicklung und sollen hier nicht weiter behandelt werden. Dem interessierten Leser liefert der „Basisreport Integration mit Web Services – Konzept, Fallstudien und Bewertung“ von Berlecon Research einen ersten Überblick. Der Basisreport, der auch die Grundlage zu wesentlichen Teilen dieses Artikels liefert, kann kostenfrei aus dem Internet bezogen werden.

An dieser Stelle sei noch einmal erwähnt, dass XML innerhalb von Web Services nicht nur als Datenformat für die Kommunikation eingesetzt wird, sondern auch die Basis für alle weiteren Web Service Standards liefert. M. a. W.: SOAP, WSDL und UDDI sind selbst XML.

Bewertung im Hinblick auf den Nutzen für Integrationsprojekte

Im Vergleich zu alternativen Integrationstechnologien unterscheiden sich Web Services vor allem durch zwei Merkmale: Standardisierung und Serviceorientierte Architektur. Im folgen-

den sollen daher die Vorteile und Herausforderungen von Web Services für Integrationsprojekte aus diesen Kriterien abgeleitet werden.

Standardisierung

Die wohl markanteste Eigenschaft von Web Services ist deren Standardisierung, die zudem auf eine breite Akzeptanz trifft: So sind die Basis-Standards selber kostenfrei zu beziehende Produkte des W3C-Konsortiums und damit per Definition herstellerunabhängig. Sie werden aber, und das ist das Besondere, bereits heute von allen namhaften Herstellern unterstützt. Zudem existiert parallel eine breite Palette von Open-Source-Produkten. Diese Unabhängigkeit von Herstellern deutet auf eine gewisse Zukunftssicherheit der Web Service – Technologie hin, die natürlich nur gegeben ist, wenn die Hersteller auch in Zukunft so agieren werden. Vieles deutet aber hierauf hin.

Die durchgängige Nutzung des XML-Standards sichert darüber hinaus erstmalig eine echte Unabhängigkeit von Programmiersprachen und Plattformen und erlaubt damit die Integration heterogener IT-Landschaften unabhängig von ihrer konkreten Ausprägung. Sie ermöglicht insbesondere eine Verbindung zwischen der .NET- und der J2EE-Welt, wie sie beispielsweise bei der Anbindung der Unteren Wasserbehörden an D-E-A (Datendrehscheibe – Einleiterüberwachung – Abwasser) vollzogen wurde.

Die Bindung an Standard-Internetprotokolle erleichtert ferner die Überwachung durch zwischengeschaltete Firewalls und trägt damit den Sicherheitsanforderungen vieler DV-Abteilungen Rechnung. Im Land NRW bedeutet dies, dass die Anwendungen der an TESTA angeschlossenen Behörden mit solchen von Landesbehörden real time integriert werden können und somit Verwaltungs- und Informationsprozesse im föderalen System transparent und steuerbar gemacht werden können.

Die Standardisierung ermöglicht darüber hinaus die Entwicklung von Codegeneratoren. So lässt sich bereits heute aus vielen Entwicklungsumgebungen bzw. ergänzenden Tools heraus WSDL und SOAP automatisch generieren und ebenso automatisiert weiterverarbeiten, was im Vergleich zu alternativen Integrationsbemühungen zu einer kostengünstigen und zügigen Projektabwicklung beiträgt. Auch ist kein herstellerabhängiges Spezialwissen nötig, was kurze Einarbeitungszeiten zur Folge haben sollte. Einschränkend muss hier allerdings hinzugefügt werden, dass der Schritt zwischen herkömmlicher, herstellerabhängiger Programmierung und dem Umgang mit Web Services nicht klein ist, da eine gänzlich andere Denkweise zugrunde liegt und entsprechende arbeitserleichternde Codegeneratoren noch nicht für alle Entwicklungsumgebungen zur Verfügung stehen bzw. noch nicht fehlerfrei funktionieren. All dies kann im konkreten Einzelfall dazu führen, dass dieser Nutzen nicht beim Pilotvorhaben, sondern erst bei Folgeprojekten transparent wird.

Generell einschränkend ist zu konstatieren, dass der XML-Schema-Standard mit einem hohen Allgemeinheitsanspruch entwickelt wurde und dementsprechend umfangreich ausgefallen ist. Dies führt dazu, dass die meisten Produkte in der Praxis nicht den gesamten Standard unterstützen. Sind bei verschiedenen Produkten die jeweils zugrunde liegenden Teilmengen des Standards aber nicht deckungsgleich, sind Kompatibilitätsprobleme vorprogrammiert. Dies gilt sinngemäß für SOAP und WSDL, da beide ebenfalls auf dem XML-Schema-Standard basieren. Dieses Problem ist schwerwiegend, da es die ursprüngliche Idee und den wesentlichen Vorteil von Web Services konterkariert. Konsequenterweise wurde mit der Web Service Interoperability (WS-I) eine Gruppe mit Vertretern der wichtigsten Softwarefirmen gegründet, die erfolgreich an der Beseitigung dieser Probleme arbeitet.

Dementsprechend redlich sind die Bemühungen dieser Firmen, die WS-I-Ergebnisse zügig in ihren Produkten zu adaptieren.

Dies hält die meisten Firmen jedoch nicht davon ab, sich am Markt zu differenzieren und so den „Preis“ als Entscheidungskriterium für die jeweilige Produktwahl auf einen untergeordneten Rang zu verweisen. Dies tun sie, indem sie ihren Produkten über den Standard hinausgehende, mehr oder weniger nützliche Features hinzufügen, deren Nutzung auf der anderen Seite die Kompatibilität zu Produkten anderer Hersteller einschränkt. Indes liegt es hier in der Entscheidung der Projektverantwortlichen, diese Features zu nutzen oder sich auf die eigentlichen Standards zu beschränken. Dies kann unter Beachtung möglicher Entwicklungsperspektiven lediglich als Kosten-Nutzen-Abwägung im konkreten Einzelfall erfolgen.

Serviceorientierte Architekturen und lose Kopplung

Unter einer serviceorientierten Architektur (SOA) versteht man zunächst ganz allgemein funktional in sich abgeschlossene, gekapselte Einheiten (Services), die miteinander kommunizieren können. Für die Nutzung eines solchen Service müssen keine Kenntnisse über dessen innere Struktur, die beliebig komplex und verflochten sein mag, vorliegen. Dies erleichtert die Wiederverwendbarkeit in verschiedenen Zusammenhängen.

Im hier beschriebenen Fall erfolgt die Kommunikation über Web Services, weshalb eine solche Integrationsarchitektur auch als Beispiel einer SOA zu verstehen ist und folglich deren Hauptvorteil, die einfache Wiederverwendbarkeit erbt. Dabei rufen sich mittels Web Services verbundene Anwendungen/Komponenten nicht direkt auf, sondern sind über standardisierte Schnittstellen (Web Services) miteinander verknüpft, also nur lose gekop-

pelt. Aufgrund dieser Eigenschaft können sich die integrierten Komponenten unabhängig voneinander ändern, solange sie die Kommunikationsschnittstelle nach außen gewährleisten. Dementsprechend einfach ist es, neue Komponenten einzubinden oder zu entfernen, sprich: Prozesse auf einer vergleichsweise abstrakten Ebene flexibel zu (re-)modellieren. Im Idealfall könnte dies bedeuten, aus einer graphisch-gestützten Fachmodellierung eines solchen Prozesses dessen technologische Umsetzung zu generieren. Mit dem Begriff „Idealfall“ ist jedoch bereits angedeutet, dass die heutige Produktwelt diese Möglichkeiten noch keineswegs derartig umfassend eröffnet und sich auch schwer tun wird, da sich noch kein einheitlicher Standard für die Workflow-Modellierung mittels Web Services durchgesetzt hat.¹⁾

Indes kommt es heute vergleichsweise selten vor, dass Anwendungen komplett neu entwickelt werden müssen. In der überwiegenden Zahl der Fälle liegen bereits Altanwendungen vor, die ehemals unter hohem Aufwand erstellt und weiter optimiert wurden und die die von ihnen zu automatisierende Aufgabe gut erfüllen. Mittels Web Services können auch solche Anwendungen uneingeschränkt zum Bestandteil einer Integrationsarchitektur werden, so dass ein gewisser *Investitionsschutz* gewährleistet wird, was eindeutig auf der Positiv-Seite der Web Services zu verbuchen ist.

Betrachtet man diese Art von Anwendungen unter dem Aspekt dynamischer Workflow-Modellierung, ist jedoch einschränkend zu bedenken, dass sie ehemals kaum unter dem Aspekt sinnvoller Serviceeinheiten designed worden sind. Zudem bieten die jeweiligen Entwicklungsumgebungen der teilweise veralteten Programmiersprachen noch nicht notwendigerweise eine ausreichende Web Service-Unterstützung an, so dass die technische Umsetzung

¹⁾ Einiges deutet darauf hin, dass BPEL4WS dieser Standard wird.

durchaus aufwändig werden kann und ggf. sogar auf die Marktpräsenz geeigneter Produkte gewartet werden muss.

Resümee

Vor einigen Jahren verlagerte sich der Fokus des betriebswirtschaftlichen Interesses von den ehemals rein funktionalen Einheiten auf Unternehmens-/Verwaltungsprozesse. Konsequenterweise galt es daher, neben den funktionalen Einheiten auch die Prozesssicht mittels DV zu unterstützen, sprich: den bereits existierenden DV-Systemen eine Prozesssichtweise hinzuzufügen. Unter dem Stichwort „Integration“ wurden in diesem Zusammenhang verschiedene Lösungsansätze entwickelt, die sich im Zuge zunehmender Globalisierung verstärkt mit der Anforderung der Flexibilität konfrontiert sahen und sehen. Web Services sind hier zunächst nicht mehr als eine (weitere) Integrati-

onstechnologie. Mit ihrer Standardisierung und SOA haben sie aber insbesondere im Hinblick auf die flexible Gestaltung von Unternehmensprozessen einige eindeutige Vorteile zu verbuchen, welche sie aus klassischen Integrationsarchitekturen deutlich hervorhebt, auch wenn diese aufgrund von Interoperabilitätsproblemen heute noch nicht vollständig zum Tragen kommen können. Insbesondere aus dem Verhalten der großen Software-Hersteller lässt sich aber prognostizieren, dass sich diese Technologie in den nächsten Jahren durchsetzen wird.

Bei all der Euphorie und den gewaltigen Vorteilen, die diese Technologie hat, darf der Einzelfall jedoch nie aus den Augen gelassen werden. So macht es meines Erachtens wenig Sinn, eine gut funktionierende Integrationsarchitektur zu Gunsten von Web Services ohne fachliche Notwendigkeit aufzuge-

ben („Investitionsschutz“) oder Schnittstellen in weitgehend homogenen, stabilen DV-Landschaften durch die Zwischenschaltung von Web Services „aufzublasen“, ohne dass sich die Vorteile dieser Technologie entfalten können.

Die zusammenfassende Antwort auf die in diesem Artikel aufgeworfene Frage lautet dementsprechend auch: Web Services sind als Technologie der Zukunft anzusehen. Dies entbindet jedoch nicht von der Entscheidung im konkreten Einzelfall, denn: Manchmal reicht klassische Integration aus.

Dr. Heike Wellmeyer
Telefon: (0211) 9449-3525
E-Mail: heike.wellmeyer@lds.nrw.de

r.z lds
Rechenzentrum im **LDS NRW**.

Unser Angebot

Sie sind interessiert, Ihre Daten auf zentralen Systemen zu speichern?

Unsere Stärken sind Ihre Vorteile:

- sicher
- performant
- skalierbar
- wirtschaftlich

Wir freuen uns auf Ihre Aufgabenstellung und unterbreiten Ihnen gern unser Angebot.

Kontakt

Speicherbereitstellung
Telefon: 0211 9449-2180
E-Mail: rechenzentrum@lds.nrw.de

Weitere Informationen finden Sie unter:
<http://lv.landesintranet.lds.nrw.de/datenverarbeitung/rechenzentrum/speicherpl/index.html>

Speicherbereitstellung im Rechenzentrum. Eine Dienstleistung des Landesamtes für Datenverarbeitung und Statistik Nordrhein-Westfalen. 2005

© LDS NRW, Düsseldorf, 2005

LDS NRW

Serverbereitstellung in der Landesdatenverarbeitungszentrale

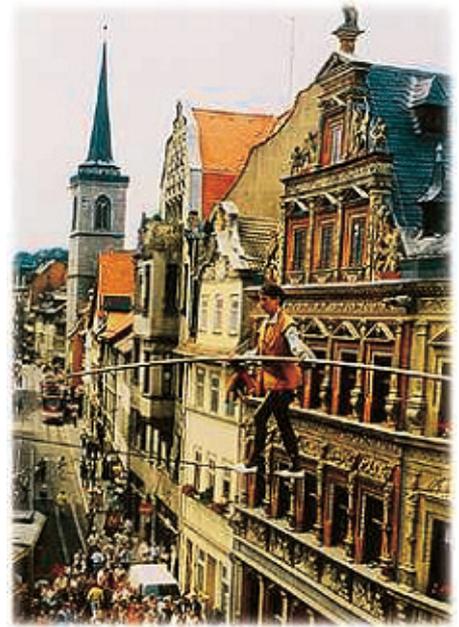
Früher war alles besser. Glauben Sie nicht? Beispiel gefällig? Na, z. B. die IT in der Landesverwaltung, die war früher umsonst. Nun, natürlich nicht wirklich umsonst, aber wenn die Genehmigung für ein Verfahren einmal erteilt war, hat man nicht mehr über Geld geredet, schon gar nicht zwischen Auftraggeber und LDS NRW. Na ja, für den Auftraggeber war das manchmal blöd, denn wer nicht zahlt, kann auch nicht meckern, falls die Leistung mal nicht seinen Vorstellungen entspricht. Eigentlich wusste damals niemand so ganz genau, was die IT-Leistungen des Landes eigentlich kosten. Unter diesem vorteilhaften Schleier ließ es sich ganz gut arbeiten bis ... , ja bis das Geld knapp wurde und die Politiker (und Bürger) verständlicherweise genauer wissen wollten, wofür das Geld der Landesverwaltung ausgegeben wird.

Daher hat sich mit dem Übergang des LDS NRW zum Landesbetrieb einiges grundlegend geändert. Die Auftraggeber des LDS NRW sind jetzt Kunden, denen Leistungen zu einem bestimmten Preis angeboten werden. Um diese Preise zu ermitteln, die auch regelmäßig vom Landesrechnungshof kontrolliert werden, hat das LDS NRW seine Kostenstruktur offen gelegt und alle relevanten Kostenfaktoren für seine Services beziffert. Bei einer solchen so genannten Vollkostenrechnung müssen alle Personal-, Sach- und Betriebskosten vollständig angegeben werden und sich in den Preisen widerspiegeln. Mit diesem Ansatz steht das LDS NRW den Kunden in der Landesverwaltung gegenüber, die i. A. keine Vollkostenrechnung machen und z. B. über die realen Kosten ihrer eigenen IT keine exakte Vorstellung haben. So werden dort dann viele Kostenfaktoren – wie z. B. Personalleistungen nach dem EhDa-Prinzip erbracht – also nach dem Motto, der Mitarbeiterin bzw. den Mitarbeiter, die/der die IT macht, müssen wir ohnehin bezahlen, also kosten ihre/seine Leistungen nichts. Hier stehen wir oft einer nicht ganz fairen Konkurrenz innerhalb der Landesverwaltung gegenüber, die auf den ersten Blick scheinbar alles billiger selber machen könnte. Bei genauerem Hinsehen relativiert sich dieses billiger meist sehr schnell.

Neben dem Kostenargument gibt es aber noch weitaus gewichtigere Gründe die zentralen Leistungen des LDS NRW einem dezentralen Hosting vorzuziehen. Um dies zu sehen, muss man sich die Leistungen des LDS NRW und die Anforderungen einer ordnungsgemäßen Datenverarbeitung etwas genauer ansehen. Dies wollen wir in diesem Artikel am Beispiel der Serverbereitstellung tun.

Anforderungen an einen sicheren IT-Betrieb

Die Bedeutung vieler Sachen, besonders der Dinge, die uns alltäglich umgeben, wird uns erst durch ihre Abwesenheit klar. Das gilt auch und gerade für so profane Dinge wie Computer. Die Konsequenzen eines Ausfalls reichen hier von bloßer Ärgerlichkeit, wenn ich auf meinem Heim-PC gerade nicht spielen kann, bis zu millionenschweren Verlusten bei Unternehmen oder gar der Gefährdung von Menschenleben, wenn z. B. ein Fahndungssystem der Polizei ausfällt. Die Abhängigkeit von dieser Technik und damit die Konsequenzen von Ausfällen steigen dabei stetig an.



Risiken richtig einschätzen

Es ist daher kein Wunder, wenn bei einer Umfrage unter Führungskräften die Verfügbarkeit der IT als bei weitem wichtigstes Qualitätsmerkmal genannt wird – vor Kosten und Leistungsfähigkeit. Dahinter steckt natürlich keine metaphysische Argumentation, sondern einfach der Aspekt, dass Ausfälle eben auch sehr hohe Kosten verursachen. Letztlich stehen wir also vor einer ökonomischen Abwägung, bei der die Kosten einer Schadensvorsorge mit dem potenziellen Schaden einer bestimmten Ausfallhäufigkeit verglichen werden müssen.

Folgende Konsequenzen hat diese Situation für einen Betreiber eines Rechenzentrums:

- Die Verfügbarkeit der angebotenen Dienstleistung ist das zentrale Qualitätskriterium.
- Die Verfügbarkeit muss in einem flexiblen Rahmen angeboten werden, der den Anforderungen des Kunden angepasst werden kann.

- Verfügbarkeit steht wie die Leistung selbst unter einem ökonomischen Vorbehalt. Daher darf Sicherheit kein Killerargument für Effizienz (Effizienz als ökonomische Bereitstellung einer wirksamen also effektiven Lösung) sein. Anders gesagt, Maßnahmen, mit denen die Verfügbarkeit erhöht wird, sollten eine definierte Ausfallsicherheit mit einem Minimum an Kosten realisieren. Dafür ist eine Analyse der Risiken unverzichtbar.
- Die Maßnahmen zur Erhöhung der Verfügbarkeit sind für den Kunden im Nicht-Schadensfall unsichtbar. Gleichzeitig ist der Kunde nicht in der Lage, die im SLA (Service Level Agreement) angegebenen Verfügbarkeitswerte zu überprüfen. Die Vermittlung dieser Informationen an den Kunden und die Herstellung des notwendigen Vertrauensverhältnisses nimmt hier also eine besonders wichtige Stellung ein; der Kunde muss betreut werden.

Serverbereitstellung im LDS NRW

Bei den Leistungen, die das LDS NRW seinen Kunden zur Verfügung stellt, steht der Sicherheitsaspekt ganz klar im Vordergrund. Da die Anforderungen der Kunden an das Sicherheitsniveau unterschiedlich sind, bieten wir dabei aufbauend auf einer soliden Grundabsicherung, die wir unter dem Stichwort geregelter Rechenzentrumsbetrieb zusammenfassen können, ein variables Konzept von Servern in verschiedenen Stufen der Ausfallsicherheit an. Dieses umfasst die im Folgenden näher beschriebenen Bereiche von Großrechnern, hochverfügbaren UNIX-Systemen und Intel-basierten Rechnern.

Neben der Frage der Bereitstellung zuverlässiger Hardware stellt der Rechenzentrumsbetrieb auch eine Support- und Wissens-Infrastruktur zur Verfügung, ohne die ein ordnungsgemäßer

Verfahrensbetrieb und eine optimale technische Konzeption nicht durchgeführt werden können.

Ein wichtiger Aspekt ist hierbei die zentrale Überwachung von Geräten und Verfahren in einer Leitstelle. Das LDS NRW verwendet dabei als Werkzeug die Software HP OPENVIEW. Agenten dieses Programms werden auf den zu überwachenden Geräten eingespielt und so konfiguriert, dass sie im Störfall automatisch eine Meldung an die Überwachungszentrale senden. Hier werden die Meldungen bearbeitet und in Trouble Tickets (das LDS NRW verwendet dabei das Trouble Ticket System REMEDY) umgewandelt, die an die für Betrieb und Betreuung der Geräte und Verfahren zuständige Stelle weitergeleitet werden. Zusätzlich dazu nimmt eine ständig verfügbare Hotline Störungsmeldungen entgegen und leitet diese ebenfalls mit Hilfe des Trouble Ticket Systems REMEDY an die Experten zu den einzelnen Verfahrensteilen weiter. Dieses Vorgehen sichert im LDS NRW eine schnelle und kompetente Behebung von Störungen.

Durch die Bündelung von IT-Fachwissen und eine stetige Weiterbildung der Mitarbeiterinnen und Mitarbeiter wird schon in der Konzeptionsphase sichergestellt, dass die vorgeschlagenen Lösungen sicher funktionieren und den Rahmen des technisch Möglichen ausschöpfen. Bei diesen personellen Ressourcen sollte im Sinne einer durchgehenden Verfügbarkeit immer mehr als eine Person als Experte für ein Gebiet zur Verfügung stehen. Die Bedeutung des personellen Aspekts für die Verfahrenssicherheit wird im Übrigen oft unterschätzt; tatsächlich geht aber die Mehrzahl der Betriebsstörungen nicht auf technische Defekte, sondern auf menschliche Irrtümer zurück.

Die Grundabsicherung des Rechenzentrums in personeller wie technischer Hinsicht schafft die Basis, ohne die auch der sicherste Server nicht zuver-

lässig arbeiten kann. Hierzu gehören neben den personellen Ressourcen die folgenden Aspekte:

- unterbrechungsfreie Stromversorgung einschließlich Notstromversorgung,
- mehrfach ausgelegte Klimatisierung,
- automatische Brandlöschanlage,
- Sicherheitstüren mit Ausweislesern, Kameraüberwachung und Berechtigungsprüfung durch das Zutrittskontrollsystem,
- räumliche Trennung der Netz- und Systemkomponenten,
- räumliche Trennung der Rechner und der zugehörigen Speicherperipherie,
- Monitorüberwachung der gesamten Einrichtung im bedienten 24-Stunden-Betrieb aus einer abgesetzten Sicherheitszentrale,
- laufende automatisierte Überwachung der Geräte und Verfahren mit automatischer Benachrichtigung im Störfall.

All diese Maßnahmen beinhalten natürlich erhebliche Investitionen. Diese lohnen sich um so eher, je größer ein Rechenzentrumsbetrieb ist. Die zwingende Konsequenz daraus ist, dass ein dezentraler Betrieb von IT-Infrastruktur entweder teurer oder weniger sicher ist als ein zentraler Betrieb.

Die Risiken, die durch eine solche Konzentration von Verfahren an einem Ort entstehen – z. B. bei einem Großbrand –, lassen sich durch Maßnahmen zur Katastrophenfallvorsorge absichern. Das Prinzip ist hierbei, Kopien der Daten oder eine zweite Server-Infrastruktur an einem hinreichend weit vom ursprünglichen Rechenzentrum entfernten Ort zu betreiben. Um dies kostengünstig realisieren zu können, arbeiten die verschiedenen zentralen Rechenzentren im Geschäftsbereich des Innenministeriums NRW zusammen und wollen diese Zusammenarbeit in Zukunft auch noch weiter ausbauen.

Großrechner-Systeme

Die Spitzenstellung im Bereich der Verfügbarkeit und Ausfallsicherheit stellen auch heute noch die Großrechner- oder genauer Mainframe-Systeme dar. Diese Rechnersysteme waren in der Anfangszeit der elektronischen Datenverarbeitung die einzigen existierenden Computer. Erst später kamen dezentrale Serversysteme – zunächst auf Basis von Unix und dann auch Windows-Betriebssystemen auf. Heute haben diese Systeme aufgrund ihrer wachsenden Leistungsfähigkeit und ihres günstigen Preises einen großen Teil der Aufgaben von den Mainframes übernommen. Dennoch gibt es eine Reihe von Anwendungen, die aufgrund ihrer Anforderungen an einen gesicherten Betrieb oder eine besonders hohe Rechnerleistung auch heute noch bevorzugt auf zentralen Mainframes betrieben werden. Das LDS NRW verwendet Systeme der Firmen Fujitsu Siemens Computer (FSC) und IBM. Diese Rechner werden unter den speziell für sie entwickelten Betriebssystemen OSD (früher BS2000) von FSC und z/OS (früher MVS) von IBM betrieben. Eine typische Anwendung für Mainframes ist z. B. das Bezügeverfahren des Landes NRW, das vom LDS NRW für das Landesamt für Besoldung und Versorgung auf dem IBM-System betrieben und weiterentwickelt wird.

Um die für dieses Verfahren notwendige Ausfallsicherheit zu gewährleisten, wurde der IBM-Mainframe in zwei Teile aufgeteilt und wird so mitsamt seiner Speicherperipherie an zwei voneinander entfernten Standorten betrieben (Geographically Dispersed Parallel Sysplex, GDPS). Hiermit ist selbst beim vollständigen Ausfall eines Standortes die Gewähr gegeben, dass der Rechnerbetrieb mit nur kurzer Unterbrechung fortgesetzt werden kann. Solche aufwändigen Verfahren der Betriebsabsicherung lohnen sich natürlich nur bei hinreichend leistungsstarken

Rechnern und damit für größere Rechenzentren. Der GDPS des IBM-Rechners stellt nur ein Beispiel für die Bedeutung von Mainframes in der heutigen IT dar.

Eine weitergehende Betrachtung dieses Themas und eine detaillierte Vorstellung der Großrechnerplattformen im LDS NRW wird in einer der nächsten Ausgaben der LDVZ-Nachrichten erscheinen.

Der baldige Tod, den man den Großrechnersystemen in den neunziger Jahren prophezeit hat, ist also offensichtlich nicht eingetreten; hier kann man eher einen gegenläufigen Trend zu einer Re-Zentralisierung des IT-Betriebes feststellen. Das liegt zum einen an den immer höher werdenden Anforderungen an die Verfügbarkeit und Sicherheit der IT, zum anderen aber auch an der immer leistungsfähiger werdenden und zunehmend auch drahtlos verfügbaren Vernetzung der Rechner. Diese allgegenwärtige Präsenz des Netzes erzeugt einen Bedarf nach Anwendungen, die zentral bereitgestellt werden und über das Netz von jedem Ort aus nutzbar sind. Eine solche – in der Endstufe mobile – Nutzung zentral betriebener Anwendungen ist nach Einschätzung von Firmen wie z. B. Microsoft die zurzeit wichtigste Veränderung auf dem IT-Sektor; und für ein zentrales Hosting von Anwendungen sind Mainframes natürlich wie geschaffen.

Dies ist sicherlich richtig, aber selbst für zentrale Anwendungen geht es auch manchmal eine Nummer kleiner (und billiger) als ein Mainframe, wie die folgenden Abschnitte zeigen sollen.

Hochverfügbare UNIX-Systeme

In den achtziger Jahren begannen die ersten UNIX Serversysteme, den etablierten Großrechnern Konkurrenz zu machen. Diese Rechner erreichten, obwohl sie kleiner und deutlich billiger

waren als die Großrechner, ähnliche Rechenleistungen. Dafür verwenden sie spezielle RISC-Prozessoren, die nur eine begrenzte Anzahl von Befehlen ausführen können, dies aber sehr schnell. Diese Prozessoren arbeiten dann mit einem maßgeschneiderten UNIX-Betriebssystem, das mit dem vorhandenen Befehlssatz zurechtkommt. Die UNIX Server wurden stetig weiterentwickelt und kommen heute in der Rechenleistung und Ausfallsicherheit den Großrechnern schon sehr nahe. Dieser Rechnertyp wird im LDS NRW vor allem für hochverfügbare und hoch performante Datenbankserver verwendet. Dabei benutzen die hier verwendeten Server das UNIX Betriebssystem SOLARIS, das ursprünglich von SUN Microsystems entwickelt wurde, aber mittlerweile auch von anderen Herstellern wie FSC unterstützt wird.

Die hohe Betriebssicherheit dieser Rechner (in den letzten Jahren gab es in diesem Bereich nicht einen einzigen ungeplanten Rechnerausfall) hat im Wesentlichen zwei Ursachen. Zum einen haben diese Geräte einen sehr hohen Grad an Produktreife sowohl im Hardware-Bereich als auch im Zusammenspiel der Hardware mit dem Betriebssystem und der Anwendungssoftware erreicht. Dieser Reifegrad und die stetige Weiterentwicklung kommen dem Anwender zugute – er zahlt aber für diese Qualität dann auch entsprechend mehr als für ein weniger aufwändig gestaltetes System auf Basis von INTEL-Prozessoren.

Der zweite Grund liegt in der Architektur der Rechner und Betriebssysteme selbst begründet, die durch eine redundante Auslegung zentraler Komponenten auch dann noch arbeitsfähig ist, wenn Teile des Systems ausfallen sollten.

Natürlich nützt auch die beste Ausfallsicherheit eines Servers wenig, wenn der verwendete Festplattenspeicher ausfällt. Aus diesem Grund werden im

LDS NRW die hochverfügbaren UNIX Server (genau wie im übrigen die Großrechner auch) nicht ausschließlich mit lokalen Festplatten betrieben, sondern speichern die Kundendaten über ein Storage Area Network (SAN) auf zentrale ebenfalls hochverfügbar ausgelegte Speichermedien. Diese Festplattensysteme von den Firmen IBM und EMC2 betreiben ihre Festplatten in einem RAID (Redundant Array of Independent Disks) Verbund. Dabei werden die Daten mehrfach auf verschiedenen Platten abgelegt, wodurch sich sowohl die Verfügbarkeit als auch die Zugriffsgeschwindigkeit der Platten enorm steigern lässt. Daneben sind in den Speichersystemen auch die Steuerungskomponenten mehrfach ausgelegt und die Platten werden laufend auf einen ordnungsgemäßen Lauf überwacht. In der Regel stellt man schon im Vorfeld fest, wenn eine Platte zu schwächeln beginnt und auszufallen droht. Diese Platten werden dann vom Hersteller des Systems im laufenden Betrieb getauscht. Ein ausführlicher Artikel über die im LDS NRW verwendeten Speichertechnologien wird ebenfalls in einer der nächsten Ausgaben der LDVZ-Nachrichten folgen.

Die Daten auf den Speichersystemen werden täglich auf Band gesichert. Wichtige Daten werden dabei automatisch in zwei verschiedene Bandspeichersysteme geschrieben, die an unterschiedlichen Orten platziert sind. Ein vollständiger Datenverlust ist damit selbst bei einem Katastrophenfall wie z. B. einem Großbrand im LDS NRW ausgeschlossen.

Intel-basierte Server

Die mit Abstand am weitesten verbreiteten Rechnersysteme weltweit basieren auf Prozessoren der Firma Intel oder sind zu diesen kompatibel. Da über 90 % aller genutzten PC-Systeme auf diesen Prozessoren aufbauen, werden von Intel-basierten Rechnern sehr hohe Stückzahlen produziert. Dadurch sind diese Systeme sehr günstig, was sich auch auf Server auswirkt, die auf dieser Architektur aufbauen. Angefeuert durch die immer schnelleren Taktraten, die ein wesentliches Verkaufsargument im PC-Markt sind, erreichen Intel-Systeme mittlerweile gute Rechenleistungen. Diesen positiven Seiten, nämlich guter Performance und niedrigem Preis, steht auf der anderen Seite das Manko einer kostenorientierten Massenproduktion entgegen. Abhängig vom konkreten Produkt ergibt sich in der Regel ein niedrigerer Qualitätslevel und eine systemtechnische Ausfallsicherheit, die nicht mit UNIX-Systemen der höheren Preisklasse oder Großrechner-Systemen vergleichbar ist.

Im LDS NRW werden Intel-Server mit den Betriebssystemen Windows und LINUX bereitgestellt. Hierbei verbuchen LINUX Systeme Vorteile in den Bereichen Performance, Betriebssicherheit und Sicherheit gegen äußere Angriffe. In einem geeigneten Umfeld, also in einem geregelten RZ-Betrieb, der auf der Grundsicherheit eines abgesicherten Rechenzentrums aufbaut, bieten solche Server eine gute Basis für den kostenbewussten Betrieb von Verfahren. Ein wichtiger Punkt hierbei ist insbesondere bei Windows-Servern das zeitnahe Einspielen von Sicherheits-Updates. Wie die jüngere Vergangenheit gezeigt hat (LoveSan Virus, Sasser

Wurm) können Unterlassungen und Verzögerungen dabei zu gravierenden Folgen führen. Die verbreitete Annahme, dass die günstigen Intel-Rechner auch von der Betreuung her besonders günstig oder einfach sind, erweist sich unter der Berücksichtigung des Sicherheitsaspekts so in der Regel als Fehlschluss. Wie oben angeführt ist gerade hier – und speziell bei den Windows-basierten Systemen – ein geregelter Informationsfluss bezüglich Sicherheitslücken und ein zeitnahe Einspielen von Updates äußerst wichtig.

Fazit

Zusammenfassend kann man sagen, dass Serverbereitstellung weit mehr umfasst, als irgendwo einen günstig erstellten Rechner aufzustellen. Die wesentlichen Kostenfaktoren liegen hier nicht in der Hardware selbst, sondern in der Schaffung gesicherter Rahmenbedingungen und der Durchführung der notwendigen Betreuungsaufgaben. Die Vernachlässigung der Aspekte Ausfall- und Datensicherheit kann im Einzelfall dramatische Konsequenzen für den Betreiber von IT-Verfahren haben, die die vordergründig eingesparten Kosten bei weitem übertreffen. Die Landesdatenverarbeitungs-zentrale berät daher ihre Kunden in Hinblick auf eine angemessene Absicherung ihrer Daten und Verfahren und stellt ihnen maßgeschneiderte Server-Lösungen zur Verfügung – vom Großrechner bis zum Windows-Server.

Dr. Michael Fluchtman
Telefon: (0211) 9449-3515
E-Mail: michael.fluchtman@lds.nrw.de

Einrichtung sicherer Verzeichnisse für Arbeitsgruppen mit Windows-Technologien

Moderne Teamarbeit erfordert häufig den Zugriff auf Dateien und Dokumente durch alle Mitglieder einer Arbeitsgruppe. Um dies zu ermöglichen, werden die Dokumente von Arbeits- bzw. Projektgruppen auf zentralen Ablage-Servern gespeichert. Arbeiten diese Gruppen mit sensiblen Informationen, so müssen die Dokumente besonders gegen unbefugten Zugriff geschützt werden. Als Beispiele seien hier Arbeitsgruppen genannt, die mit personenbezogenen Daten arbeiten oder Arbeitsgruppen, die strategische Planungen für ihre Organisation durchführen. In diesen Fällen ist es angeraten, die Dokumente nicht nur durch Zugriffsberechtigungen zu schützen, sondern zusätzlich durch Verschlüsselung das Lesen der Informationen beispielsweise durch Systemadministratoren oder während des Datentransfers durch lokale Netze zu verhindern. Bei der Nutzung einer Verschlüsselungstechnik muss aber der Zugriff auf die Daten durch alle berechtigten Mitglieder der Arbeitsgruppe sichergestellt werden, d.h. jedes Mitglied muss in die Lage versetzt werden, die Daten wieder zu entschlüsseln, weiterzubearbeiten und anschließend wieder verschlüsselt abzulegen.

Problembeschreibung

Die prinzipielle Aufgabenstellung soll nun im folgenden Abschnitt anhand eines kleinen Beispiels illustriert werden. In Abbildung 1 ist die Infrastruktur einer kleinen Organisationseinheit bestehend aus zwei Arbeitsgruppen dargestellt.

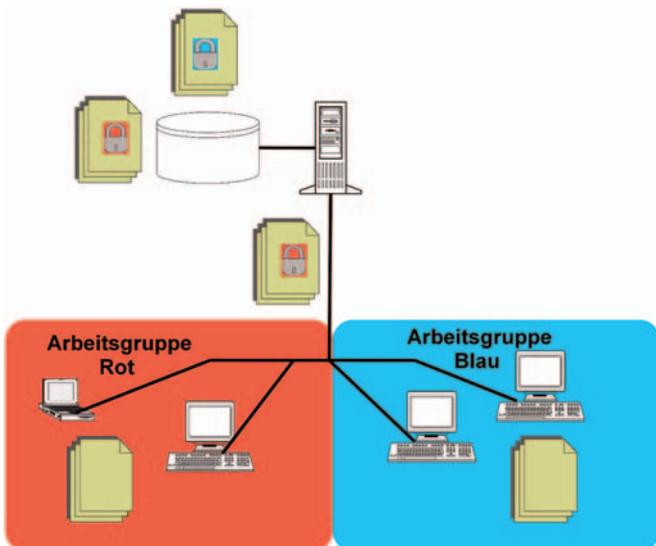


Abb. 1: Infrastruktur einer kleinen Organisationseinheit

Die beiden Gruppen „Rot“ und „Blau“ bestehen jeweils aus mehreren Mitgliedern, die gemeinsam an einer Aufgabe arbeiten. Die Daten, die von beiden Arbeitsgruppen benötigt werden, sind auf einem gemeinsam genutzten File Server abgelegt. Das Netzwerk, über das die Arbeitsplatz-PCs der Mitglieder beider Gruppen mit dem File Server verbunden sind, wird ebenfalls von beiden Arbeitsgruppen genutzt. Somit muss in dieser Umgebung sichergestellt sein, dass die den Arbeitsgruppen zugeordneten Dokumente jeweils ausschließlich von den Mitgliedern der Arbeitsgruppe bearbeitet werden können. Dabei ist das Lesen der Dokumente bereits als Bearbeitungsschritt anzusehen. Somit reicht der einfache Zugriffsschutz über das Dateisystem nicht aus, da in diesem Fall sowohl die Administratoren des File Servers als auch die Netzwerkadministratoren die Inhalte der Dokumente lesen könnten. Daher müssen alle Dokumente der beiden Arbeitsgruppen sowohl auf dem Dateisystem des File Servers als auch auf dem Transport zwischen File Server und den Arbeitsplatzrechnern der Mitarbeiterinnen und Mitarbeiter verschlüsselt werden. Lediglich auf den Arbeitsplatz-PCs dürfen die Dateien in Klarschrift vorliegen.

Somit ergibt sich eine Struktur entsprechend Abbildung 2. Unter dem gemeinsamen Verzeichnis „Arbeitsgruppen“ sind die beiden Unterverzeichnisse „Blau“ und „Rot“ angelegt.

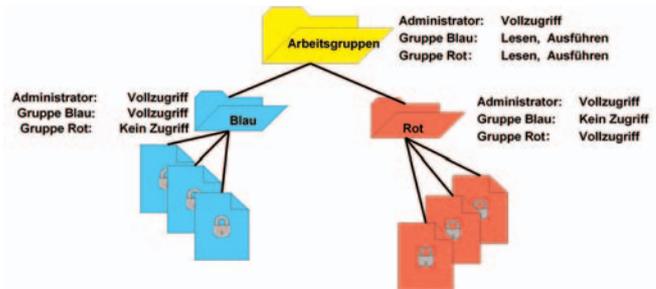


Abb. 2: Zugriffsrechte auf die Verzeichnisse der Arbeitsgruppen

Die Zugriffskontrolle wird über Standardmechanismen des Dateisystems geregelt. Dies bedeutet, dass die Systemadministratoren Vollzugriff auf alle Verzeichnisse und somit auch auf alle Dokumente in den Verzeichnissen haben. Die Mitarbeiter der Arbeitsgruppe „Blau“ haben die Rechte, die notwendig sind, um über den Einstiegsordner „Arbeitsgruppen“ auf ihre Dokumente in ihrem Verzeichnis „Blau“ zuzugreifen. Auf den Ordner „Blau“ haben Sie Vollzugriff, um dort Dateien zu lesen, zu verändern, zu löschen und neu anzulegen. Auf das Verzeichnis „Rot“ haben Sie keinerlei Zugriffsrechte. Die Mitarbeiterinnen und Mitarbeiter der Arbeitsgruppe „Rot“ haben die entsprechenden Rechte für ihr

Verzeichnis und keine Rechte für den Ordner „Blau“. Somit ist der wechselseitige Ausschluss der Mitglieder der Arbeitsgruppen auf die Dokumente der jeweils anderen Arbeitsgruppe geregelt. Um auch den Zugriff auf die Dateien der Arbeitsgruppen durch die Administratoren zu unterbinden, werden die Dateien durch die Mitglieder der Arbeitsgruppen verschlüsselt.

Hierfür gibt es verschiedene Möglichkeiten, so dass nachfolgend ein Überblick über Verschlüsselungstechniken gegeben wird.

Verschlüsselung

Bei der Verschlüsselung von Daten werden diese in eine unleserlich erscheinende Form umgewandelt. Ziel dabei ist es, dass nur berechtigte Personen die Daten wieder in die ursprüngliche Form bringen können. Sie kennen die zugehörige Entschlüsselungsvorschrift. Das „Knacken“ der Verschlüsselung durch Unbefugte ohne Kenntnis dieser Vorschrift sollte praktisch unmöglich sein.

Kryptographie, die Wissenschaft des Verschlüsseln, war lange Zeit ausschließlich Sache der Militärs. Heute, wo Homebanking und Online-Shopping an der Tagesordnung sind, gibt es ein breites Bedürfnis an verschlüsselten Übertragungen und Speichern von Informationen. Moderne Verschlüsselungstechniken setzen Algorithmen ein, die neben dem zu chiffrierenden Klartext als weiteren Parameter einen Schlüssel benötigen. Nur mit einem passenden Schlüssel, einer Art Passwort, gelangt man vom Klartext zum verschlüsselten Geheimtext und umgekehrt. Ein grundlegendes Prinzip der Kryptographie sagt aus, dass die Sicherheit einer verschlüsselten Information vom Schlüssel abhängt, da der genutzte Algorithmus bekannt ist.

Symmetrische Verschlüsselungsverfahren

Bei symmetrischen Verschlüsselungsverfahren benötigt man sowohl zum Verschlüsseln als auch zum Entschlüsseln denselben Schlüssel. Wird eine Information verschlüsselt verschickt, müssen sich Sender und Empfänger vorher auf einen Schlüssel geeinigt haben. Normalerweise ist solch ein Schlüssel wesentlich kürzer als die Nachricht selber und kann mehrmals verwendet werden.

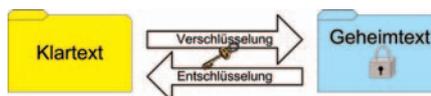


Abb. 3: Symmetrische Ver- und Entschlüsselung mit demselben Schlüssel

Symmetrische Verschlüsselungsverfahren können unzureichend sein. So wurde beispielsweise bei dem Textverarbeitungsprogramm „WordPerfect 5“ eine Option zum verschlüsselten Abspeichern von Texten mit einem nicht öffentlich gemachten Verfahren angeboten. Mit Sicherheit hatte das wenig zu tun, da schon im 19. Jahrhundert entsprechende Methoden zum Knacken solcher Chiffren entwickelt wurden.

Bei gutem Design eines symmetrischen Verfahrens erhält man neben hoher Sicherheit auch hohe Effizienz beim Ausführen auf Computern, da vor allem mit Verschiebungen, Vertauschungen und der XOR-Operation hardwarenah auf Bit-Ebene gearbeitet wird. Als wichtigster Vertreter symmetrischer Algorithmen sei der DES (Data Encryption Standard) genannt, der von den 70er- bis in die Mitte der 90er-Jahre zum Verschlüsseln in der US-amerikanischen Verwaltung genutzt wurde. Allein die Tatsache, dass Schlüssellängen auf 56 Bit begrenzt waren und es mit heutigen PC-Verbänden in relativ kurzer Zeit möglich ist alle 2^{56} (= 72 Billionen) möglichen Schlüssel durchzuprobieren, machte die Ablösung von DES notwendig. Nach einem langen Auswahlverfahren wurde der Algorithmus Rijndael der

Belgier J. Daemen und V. Rijmen im Jahr 2000 zum neuen AES (Advanced Encryption Standard) auserkoren. AES arbeitet mit wahlweise 128, 192 oder 256 Bit als Schlüssellänge. Dabei bedeutet jedes zusätzliche Bit eine Verdopplung des Aufwands, alle möglichen Schlüssel durchzutesten.

Public Key-Verschlüsselung

Der Nachteil eines symmetrischen Verfahrens ist, dass man mit jedem Kommunikationspartner einen eigenen Schlüssel vereinbaren muss, denn nutzt man einen Schlüssel mehrfach mit verschiedenen Partnern, so kann jeder von ihnen jede Nachricht an einen aus dieser Gruppe entschlüsseln. Somit erfordern symmetrische Verfahren einen höheren Verwaltungsaufwand, da zudem sichergestellt werden muss, dass jeder Schlüssel geheim bleibt, also vor unbefugtem Zugriff geschützt ist.



Abb. 4: Public Key-Verschlüsselung mit zwei verschiedenen Schlüsseln

Um diesen Mangel zu beheben, werden Public Key-Verfahren (asymmetrische Verfahren) verwendet. In solch einem Szenario besitzt jeder Teilnehmer zwei Schlüssel, einen öffentlichen und einen privaten. Allein der private Schlüssel muss geheim gehalten werden. Der öffentliche Schlüssel wird dagegen in einer Art „Telefonbuch“ verbreitet. Will man nun jemandem eine geheime Botschaft zukommen lassen, so holt man sich aus diesem Buch den öffentlichen Schlüssel desjenigen und verschlüsselt damit die Nachricht. Allein der Empfänger kann nun mittels des privaten Schlüssels die Nachricht dechiffrieren.

Public Key-Verschlüsselung ist erst seit den 80er-Jahren bekannt. R. Rivest, A. Shamir und L. Adleman entwickelten bei dem Versuch nachzuweisen, dass es so etwas wie asymmetrische Verfahren nicht geben kann,

genau solch einen Algorithmus, den RSA-Algorithmus. Dieser ist heutzutage noch immer ein wichtiger Vertreter asymmetrischer Kryptographie. Heute findet man Public Key-Verfahren in HBCI (zum Homebanking), SSL (für das https-Protokoll) oder PGP.

Public Key-Verschlüsselung eröffnet zudem die Möglichkeit einer digitalen Unterschrift, für die prinzipiell nur die Anwendungsreihenfolge von öffentlichem und privatem Schlüssel geändert wird. Der Ersteller einer Botschaft signiert diese, indem er die Nachricht zusammen mit der von ihm durch den privaten Schlüssel „verschlüsselte“ Nachricht versendet. Jeder kann den letzten Teil nun mit dem öffentlichen Schlüssel des angeblichen Senders entschlüsseln und mit dem mitgelieferten Klartext vergleichen. Bei Übereinstimmung wird das Dokument als vom Sender erstellt akzeptiert.

Die Kombination beider Vorgehensweisen

Ein Nachteil asymmetrischer Verfahren ist die hohe Rechenleistung, da im Gegensatz zu symmetrischen Verfahren kompliziertere Funktionen ausgeführt werden müssen. Doch eine Kombination beider Verschlüsselungsarten schafft Abhilfe. Ein Klartext wird mit Hilfe eines symmetrischen Verfahrens verschlüsselt. Der Schlüssel kann dabei beliebig gewählt werden. Will man den Text nun einem anderen zukommen lassen, so verschlüsselt man zunächst den Schlüssel für das symmetrische Verfahren mit dem öffentlichen Schlüssel des Empfängers aus einem Public Key-Verfahren. Der verschlüsselte Schlüssel sowie die verschlüsselte Botschaft werden dann versendet. Der Empfänger kann mit Hilfe seines privaten Schlüssels den ursprünglichen Schlüssel dechiffrieren und damit die Botschaft lesbar machen.

Der Vorteil dieser Kombination ist, dass sich Sender und Empfänger nicht im Vorfeld auf einen Schlüssel für ein symmetrisches Verfahren geeinigt haben müssen. Das Public Key-Verfahren wird außerdem nur für einen relativ kleinen Text, nämlich den Schlüssel des symmetrischen Verfahrens, benötigt. Somit werden die Vorteile beider Verfahren in dieser Vorgehensweise vereint.

Umsetzung mit Windows-Technologien

Es gibt prinzipiell zwei Möglichkeiten, die oben genannten Anforderungen an eine Verschlüsselungslösung für Arbeitsgruppen zu erfüllen. Beide basieren auf der seit Windows 2000 im Dateisystem verankerten Windows-Technologie Encrypting File System (EFS). Unterschiede ergeben sich in der Art und Weise, wie die Daten über das Netzwerk transportiert und auf dem Server gespeichert werden.

Encrypting File System (EFS)

EFS nutzt sowohl das symmetrische Verschlüsselungsverfahren AES als auch ein Public Key-Verfahren mit einem öffentlichen und einem privaten Schlüssel. Der öffentliche Schlüssel des Public Key-Verfahrens wird im Verzeichnisdienst Active Directory an das Benutzerkonto gebunden. Der private Schlüssel verbleibt auf dem Rechner des Anwenders in einem besonders geschützten Bereich des Benutzerprofils. Der Zugang zu den Daten ist nur über das jeweilige Benutzerkonto und den Rechner möglich.



Dateien werden mittels AES ver- und entschlüsselt. Für jede Datei wird dazu ein Schlüssel mit der Länge von 256 Bit generiert, der mit dem öffentlichen Schlüssel des Anwenders chiffriert an

die Datei angehängt wird. Sollen weitere Benutzer Zugang zu dieser Datei erhalten, so wird nach demselben Muster der entsprechend chiffrierte Schlüssel an die Datei angefügt. Es besteht keine Notwendigkeit für die Einrichtung eines administrativen Wiederherstellungsagenten (Key Escrow).

SMB + IPsec + EFS

Das gebräuchlichste Protokoll zum Austausch von Dateien im Windows-Netzwerk ist das proprietäre Server-Message-Block-(SMB-)Protokoll. Kombiniert man SMB und EFS, so findet die Verschlüsselung der Dateien auf dem Server statt. Zur Erfüllung der Anforderung, dass auch Netzwerkadministratoren keinen Zugriff auf die Daten haben, müssen die Daten beim Transport über das Netzwerk zusätzlich verschlüsselt werden. Hierzu kann das gebräuchliche Protokoll IPsec verwendet werden.

WebDAV + EFS

Web Distributed Authoring and Versioning (WebDAV) wurde herstellerübergreifend als Standard für den Austausch von Dateien über das Internet entwickelt. Dank der engen Integration in Windows kann es das SMB-Protokoll in jeder Hinsicht ersetzen. Kombiniert man WebDAV und EFS, so findet die Verschlüsselung auf dem PC des Anwenders statt. Die Übertragung und Speicherung der verschlüsselten Daten geschieht transparent. Die Verwendung eines Protokolls, das den Datentransport zusätzlich sichert, kann somit entfallen.

Zusammenfassung

Unter Einsatz der beiden Windows-Technologien Encrypting File System (EFS) und Web Distributed Authoring and Versioning (WebDAV) kann eine Verschlüsselungslösung für Arbeits-

gruppen mit Windows-Bordmitteln bereitgestellt werden. In Kombination erfüllen diese Technologien folgende Anforderungen:

- Die Dokumente können von allen Mitgliedern einer Arbeitsgruppe bearbeitet werden.
- Die Ver- und Entschlüsselung findet auf dem Rechner des Nutzers statt.
- Die Daten werden verschlüsselt über das Netzwerk transportiert.
- Die Daten werden verschlüsselt auf einem Server gespeichert.

Somit ist es möglich, ohne den Einsatz weiterer kostenpflichtiger Software, Dokumente mit vertraulichem Inhalt zu schützen. Das IT-Service-Center des LDS NRW hat mit Hilfe der beschriebenen Technologie erfolgreich verschlüsselte Arbeitsbereiche für Personaldaten verarbeitende Arbeitsgruppen im LDS NRW eingerichtet. Falls auch in Ihrer Organisation der Bedarf nach einer solchen Lösung besteht, wenden Sie sich bitte an einen der Autoren.

Stefan Willer
Telefon: 0211 9449-2252
E-Mail: stefan.willer@lds.nrw.de

Dipl.-Math. Holger Traczinski
Telefon: 0211 9449-2123
E-Mail: holger.traczinski@lds.nrw.de

Dipl. Inform. Dr. Jörg Hintelmann
Telefon: 0211 9449-2424
E-Mail: joerg.hintelmann@lds.nrw.de

rz lds
Rechenzentrum im **LDS NRW.**

Ihre Vorteile

Sie sind daran interessiert, Ihre IT-Verfahren unter Einbeziehung von server- und datenbezogenen K-Fall-Vorsorgemaßnahmen im Rechenzentrum des Landesamtes für Datenverarbeitung und Statistik betreiben zu lassen?

Wir bieten folgende IT-Dienstleistungen in diesem Produktsegment an:

- serverbezogene K-Fall-Sicherung
- datenbezogene K-Fall-Sicherung
- regelmäßige K-Fall-Übungen
- Erstellung von K-Fall-Anweisungen
- K-Fall-Maßnahmen im Druckbetrieb
- K-Fall-Absicherung an verschiedenen Standorten

Wir freuen uns auf Ihre Aufgabenstellung und unterbreiten Ihnen gern unser Angebot.

Kontakt

K-Fall-Vorsorgemaßnahmen
Telefon: 0211 9449-2396
E-Mail: rechenzentrum@lds.nrw.de

Weitere Informationen finden Sie unter:
http://lv.landesintranet.lds.nrw.de/datenverarbeitung/rechenzentrum/LINKVERZEICHNIS/RZ_Services_K_Fall_Druck.html

K-Fall-Vorsorgemaßnahmen im Rechenzentrum. Eine Dienstleistung des Landesamtes für Datenverarbeitung und Statistik Nordrhein-Westfalen. **2005**

© LDS NRW, Düsseldorf, 2005

Mono:

.Net für Linux

Was ist Mono? In den Worten der Entwickler (www.mono-project.com) ist Mono eine „umfassende Open Source-Entwicklungsplattform, basierend auf dem .Net-Framework, die es Entwicklern erlaubt, Linux-Anwendungen und plattformübergreifende Anwendungen mit nie da gewesener Produktivität zu erstellen“. Man könnte auch sagen, Mono ist eine .Net-Portierung für Linux.

.Net

Das von Microsoft u. a. als Konkurrenz zu Java entwickelte .Net-Framework (<http://www.microsoft.com/Net/Basics.aspx>) erfreut sich seit seiner Einführung im Jahr 2000 einer wachsenden Beliebtheit – ermöglicht es doch die komfortable Erstellung von GUI-Anwendungen (Graphical User Interface), Webanwendungen, Web Services, Klassenbibliotheken, ActiveX-Steuer-elementen und vielen anderen Anwendungstypen. Dabei erfolgt die Entwicklung wahlweise in einer der Sprachen C#, Visual Basic, C++, Visual J# oder einer von weiteren Sprachen, für die .Net-Compiler von Drittanbietern existieren, jedoch mit einer einheitlichen Klassenbibliothek für alle Sprachen.

Ähnlich wie schon bei der Vorgängerversion der Sprache Visual Basic ist ein zentraler Punkt hier das Thema „Produktivität“: Tatsächlich ist die Entwicklung mit .Net um einiges unkomplizierter und damit oftmals schneller als z. B. mit C++ und gelingt auch weniger erfahrenen Programmierern – ohne dass man dabei auf die Merkmale einer vollwertigen objektorientierten Programmiersprache verzichten müsste. Diese Einfachheit gilt auch für die mit .Net vorgestellte neue Sprache C#, die mit ihrer C-ähnlichen Syntax eher C++- und Java-Programmierer anspricht. Selbst im Vergleich mit Java schneidet .Net auf Grund der Verfügbarkeit leistungsfähiger Programmierwerkzeuge gut ab. Die Originalimplementierung von Microsoft ist allerdings auf Windows-Betriebssysteme beschränkt.

... für Linux

Mit der Standardisierung des .Net-Frameworks und der Sprache C# hat Microsoft selbst die Möglichkeit geschaffen, dass auch andere Hersteller Implementierungen anbieten. Das betrifft einerseits natürlich die Entwicklung von Compilern. So gibt es etwa auch leistungsfähige Entwicklungswerkzeuge für .Net von Borland, und auch Mono bringt ei-

nen eigenen C#-Compiler und einen Compiler für VB.Net mit. Compiler für andere Sprachen befinden sich in der Entwicklung.

Auf Grund der Tatsache, dass .Net-Programme nicht unmittelbar auf dem Betriebssystem, sondern auf einer Laufzeitumgebung aufsetzen, ist es andererseits aber auch möglich, durch eine Portierung dieser Laufzeitumgebung solche Programme auch auf anderen Betriebssystemen ablauffähig zu machen (und zwar ohne Neukompilierung). Eine solche Portierung der Laufzeitumgebung (Common Language Runtime) stellt das Mono-Projekt zur Verfügung.

Nun besteht .Net aber nicht ausschließlich aus der Common Language Runtime, sondern der zweite wichtige Bestandteil ist eine Klassenbibliothek (Framework Class Library), die von allen .Net-Sprachen gemeinsam genutzt werden kann. Ein weiterer wesentlicher Bestandteil von Mono ist deshalb eine eigene, neu programmierte Version der Klassenbibliothek. Zusätzlich dazu gibt es noch eine ganze Reihe von Bibliotheken, die etablierte Linux-Funktionalität für Mono und damit für die .Net-Programmierung verfügbar machen (z. B. gtk+, GNOME-Bibliotheken, Glade).

Last not least enthält Mono außerdem eine eigene Open Source-Entwicklungsumgebung, (MonoDevelop), basierend auf dem auch in Windows-Entwickler-Kreisen beliebten, selbst in C# geschriebenen SharpDevelop.

Open Source-Project

Einer der Schlüsselbegriffe in der eingangs zitierten Selbstbeschreibung des Mono-Projekts ist sicherlich „Open Source“. Im Gegensatz zum Microsoft-Original ist die Mono-Implementierung der Klassenbibliothek und auch des C#-Compilers quelloffen.

Somit ist Mono auch keineswegs auf Linux beschränkt, sondern Portierungen auf weitere Betriebssysteme existieren oder werden entwickelt.

Rückkehr nach Monkey Island

Warum der Name Mono? Mono ist das spanische Wort für „Affe“, was eigentlich auch nichts erklärt, aber vermutlich daher rührt, dass der Mono-„Vater“ Miguel de Icaza – ein

bekannter Open Source-Entwickler und Mitbegründer des Gnome-Projekts – mexikanischer Herkunft ist. Das Mono-Handbuch wird denn auch von der Fangemeinde liebevoll als „Monkeyguide“ bezeichnet.

Wer entwickelt Mono?

Ist nun Mono lediglich eine Spielerei einer Hand voll jugendlicher Programmierfreaks, oder steckt dahinter auch eine seriöse Firma mit dem entsprechenden Potenzial für die Weiterentwicklung des Projekts? Die Entwickler-Firma Ximian hatte sich im Linux-Umfeld bereits einen Namen gemacht, bevor sie von Novell aufgekauft wurde. Und dass Novell diese Initiative unterstützt, kann sicher ein Zeichen für die längerfristige Perspektive des Vorhabens gesehen werden. Novell verwendet Mono bereits in eigenen Projekten, und Mono wird in den Novell-eigenen, SuSE-basierten Linux-Desktop integriert sein – so wie ja auch Microsoft beabsichtigt, das .Net-Framework in künftige Betriebssystemversionen zu integrieren.

Lizenzfragen

Trotz dieser günstigen Aussichten wird Mono bislang nur zögerlich in kommerziellen Projekten eingesetzt. Das mag einerseits damit zu tun haben, dass noch nicht alle Features des Windows-Gegenstücks für die Linux-Version implementiert sind (s.u.), andererseits stellen sich einige Anwender die Frage, ob Microsoft dem Mono-Projekt auch dann noch wohlwollend gegenüberstehen wird, wenn dieses sich als ernst zu nehmende Konkurrenz zur Microsoft-Implementierung herausstellt und somit zum Erfolg des Linux-Betriebssystems beiträgt. Es wird gefragt, ob Microsoft in diesem Fall lizenzrechtlich gegen die Mono-Entwickler vorgehen könnte. Das Mono-Team selbst weist solche Betrachtungen aber zurück – man sieht sich rechtlich auf sicherem Boden.

Alternativen

Tatsächlich hat Microsoft ja möglicherweise auch ein eigenes Interesse an einer Unix-Version des Frameworks: gerade die Plattformabhängigkeit wird .Net ja (z. B. im Vergleich mit Java) häufig als Nachteil angekreidet. So unterstützt denn Microsoft mit dem Projekt „Rotor“ auch eine eigene Unix-Implementierung – allerdings nicht für Linux, sondern für FreeBSD. Diese ist außerdem ausdrücklich für Lehr- und Forschungszwecke, nicht jedoch für den produktiven Einsatz gedacht.

Java

Bei aller Begeisterung für die Einfachheit, mit der dank Mono jetzt Windows-Entwickler auch Linux-Anwendungen – sogar mit grafischer Benutzeroberfläche – programmieren können, sollte eines nicht vergessen werden: Mit Java ist dies schon seit Jahren möglich. Mit der Java Virtual Machine steht ebenfalls eine – schon immer plattformunabhängige – Laufzeitumgebung zur Verfügung, welche es ermöglicht, dieselbe Anwendung ohne Neukompilierung sowohl unter Windows als auch unter verschiedenen Unix-Derivaten (natürlich auch Linux) auszuführen.

Features und Einschränkungen

Wie vollständig ist nun die .Net-Umsetzung von Mono? Sind tatsächlich alle Programme uneingeschränkt ablauffähig? Die Vollständigkeit ist tatsächlich erstaunlich, selbst in Bereichen wie Webanwendungen, wo die Microsoft-Version einen engen Bezug zum Betriebssystem aufweist. Dennoch gibt es auch einige Einschränkungen. Beides soll im Folgenden näher beleuchtet werden.

ASP .Net

Mit ASP (Active Server Pages) .Net wird derjenige Teil der Klassenbibliothek bezeichnet, der für die Entwicklung von Webanwendungen verwendet wird. Dazu muss natürlich der Webserver ASP.Net unterstützen. Unter Windows ist dies Microsofts Webserver, der Internet Information Server. Dieser steht natürlich unter Linux nicht zur Verfügung. Mono bringt deshalb einen eigenen kleinen Webserver (XSP) mit. Aber auch für den Linux-Haus-und-Heim-Webserver Apache gibt es bereits ein Modul, das diesen um ASP.Net-Unterstützung erweitert.

Windows Forms/GTK#

Einer der am schwersten für Linux umzusetzenden Teile von .Net ist die Unterstützung von Programmen mit grafischer Benutzeroberfläche – in der Microsoft-Klassenbibliothek durch die Windows-Forms-Klassen realisiert. Da die Fensterverwaltung unter Linux grundsätzlich anders organisiert ist als unter Windows, ist eine 1:1-Übertragung mit Problemen verbunden. Während die Umsetzung von Windows Forms für Mono noch relativ neu ist, steht Entwicklern mit GTK# bereits seit längerer Zeit eine (Mono-eigene) Klassenbibliothek für die Entwicklung von GUI-Programmen zur Verfügung.

Serviced Components

Für einen Bestandteil von .Net gibt es allerdings noch kein Linux-Äquivalent, und dies ist auch nicht geplant: Mit Serviced Components ermöglicht die Windows-Version von .Net den Zugriff auf COM+-Dienste und damit auf Infrastrukturdienste des Betriebssystems wie z. B. verteilte Transaktionen mit dem Microsoft Transaction Server. Diese Technologie lässt sich noch an ehesten mit den Enterprise

Java Beans (EJB) von Java (J2EE, Java 3 Enterprise Edition) vergleichen. Um dasselbe unter Linux zu implementieren, müsste entweder das Linux-Betriebssystem ebenfalls um COM+-Dienste erweitert werden, oder die Komponenten müssten, wie im Java-Umfeld üblich, in einer Art „Application Server“ gehostet werden. Wahrscheinlicher ist aber, dass hier auf die genaue Nachbildung der Windows-Lösung verzichtet und stattdessen für entsprechende Aufgabenstellungen Unix-eigene Mittel eingesetzt werden.

Was ist Mono nicht

Microsofts Marketing hat den Begriff „Net“ verquickt mit einer Reihe anderer Technologien wie Server-Betriebssystem und -Anwendungen, Passport/Hailstorm, Software-as-a-Service etc. Mono hat mit diesen Technologien nichts zu tun: Mono ist eine Implementierung von .Net im Sinne einer Entwicklungsplattform, bestehend aus der Laufzeitumgebung, Compilern und Klassenbibliotheken.

Fazit

Mono ermöglicht es Entwicklern, mit wenig Aufwand Anwendungen zu

erstellen, die sowohl unter Windows als auch unter Linux ablauffähig sind. Damit wird gerade auch Windows-Entwicklern, die in C# oder Visual Basic.Net programmieren, die Linux-Welt erschlossen. In einigen mittelgroßen Projekten wird Mono bereits produktiv eingesetzt. Es lohnt sich also sicherlich, diese in der Entwicklung befindliche Technologie im Auge zu behalten.

Dr. Susanne Wigard

Telefon: 0211 9449-6013

E-Mail: susanne.wigard@lds.nrw.de



Rechenzentrum im **LDS NRW**.

Ihre Vorteile
Sie sind interessiert, Ihre Druckaufträge im Rechenzentrum des Landesamtes für Datenverarbeitung und Statistik ausführen zu lassen?

Für die Erstellung Ihrer Produkte – von der Beratung bis zum Versand – bieten wir Ihnen eine umfassende Dienstleistung im

IT-Druck • Offsetdruck • Kopierservice an.

Unsere Stärken sind Ihre Vorteile
im IT-Druck Service von den Produktionsplattformen IBM/z/OS – FSC/BS2000/OSD – Windows NT/XP

Druckausgabe auf

- Endlos- oder Einzelblatt im Simplex- oder Duplexverfahren
- DIN A4, A3

im Digital-/Offsetdruck und Kopierservice

- Text- und Layouterstellung
- Ein- bis Vierfarbendruck
- Buchbinderische Verarbeitung

Fertigung von Vorlagen oder Dateien

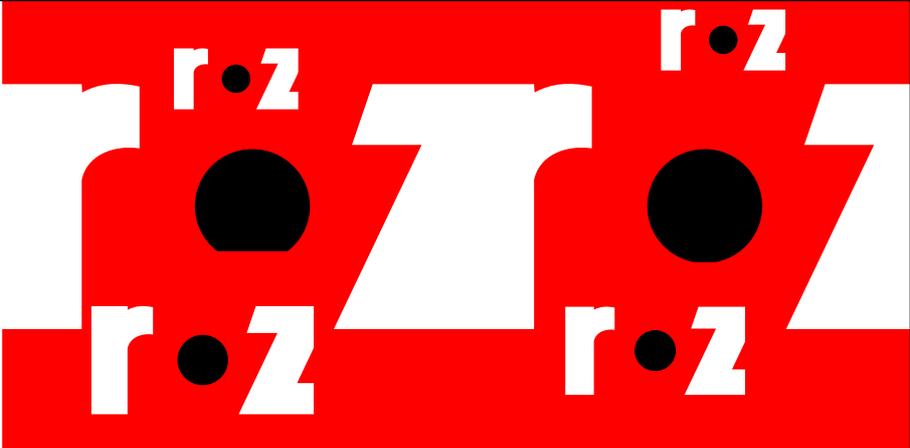
- Formulare, Prospekte, Broschüren, Flyer
- Serienbriefe
- Visitenkarten
- Kopien

Sonderanwendungen wie

- Eindruck in vorgedruckte Formulare
- Personalisierung
- Scannen von Vorlagen inkl. Bereitstellung über das Netzwerk als PDF-Dokument

ergänzen das Spektrum Ihrer Anforderungen.

Wir freuen uns auf diese Aufgabenstellung und unterbreiten Ihnen gern unser Angebot.



Druckerzeugnisse im Rechenzentrum.
Eine Dienstleistung des Landesamtes für Datenverarbeitung und Statistik Nordrhein-Westfalen. 2005

Kontakt

IT-Druck:
Telefon: 0211 9449-2651
E-Mail: rechenzentrum@lds.nrw.de

Offsetdruck und Kopierservice:
Telefon: 0211 9449-3515
E-Mail: rechenzentrum@lds.nrw.de

Weitere Informationen finden Sie unter:
http://lv.landesintranet.lds.nrw.de/datenverarbeitung/rechenzentrum/LINKVERZEICHNIS/RZ_IT_Druck.html

© LDS NRW, Düsseldorf, 2005



WAVE

Eine umfassende Unterstützung für die Veranstaltungsplanung und -durchführung

Koordination und Kommunikation sind die beiden wichtigsten Aufgaben der Vertretungen des Landes Nordrhein-Westfalen beim Bund und bei der EU. Die Protokollreferate der Vertretungen des Landes NRW in den Außenstellen Berlin und Brüssel führen im Jahr ca. 50 Großveranstaltungen durch. Hierzu werden die Stammdaten von bis zu 10 000 Personen des öffentlichen Lebens verwaltet. Je nach Art der Veranstaltung werden aus diesem Personenkreis bis zu 2 000 Personen eingeladen. Hierbei muss ein besonderes Augenmerk auf die protokollarisch korrekte Anrede gerichtet werden. Die einzuladenden Personen erhalten Einladungskarten und Rücklaufkarten, auf denen der Teilnahmewunsch vermerkt wird sowie anschließend Einlasskarten, die bei der Veranstaltung abgegeben werden.

Zur Unterstützung dieser Aufgaben wurde WAVE (Windows Adress-Veranstaltungs-Einladungsverwaltung) entwickelt. Es ist eine typische Windows-Applikation, die eine einfache und intuitive Bedienung erlaubt.

WAVE unterstützt folgende Aufgabenbereiche:

- Verwaltung von Adressangaben und weiteren Informationen zu Personen (Personenstammdaten)
- Verwaltung sog. Funktionskreise
- Verwaltung von Veranstaltungen
- Veranstaltungsplanung und -durchführung
- Serienbrief- und Serien-E-Mail-Erstellung
- Drucken von Listen und Etiketten
- Statistik über durchgeführte Veranstaltungen



Abb. 1: WAVE-Hauptmenü

Personenstammdaten

Hier werden die eigentlichen Personendaten erfasst bzw. geändert. Zu beachten ist dabei, dass eine Person mehrere Adressen haben kann, z. B. Privatadresse und verschiedene dienstliche Adressen. Diese Adressen korrespondieren mit den unterschiedlichen Funktionen einer Person, d. h. einer Person werden verschiedene Funktionskreise zugeordnet (siehe auch nächster Abschnitt), über die sie dann später zu Veranstaltungen eingeladen werden kann. Zu den einzelnen Adressen der Personen können individuelle Anreden für die Serienbriefe erfasst werden. Personen können auch Institutionen zugeordnet werden, um die Erfassung der Adressdaten zu erleichtern. Selbstverständlich stehen auch umfangreiche Suchmöglichkeiten zur Verfügung. So ist es möglich, neben den Personendaten auch nach allen Adressdaten und nach Funktionskreisen zu suchen.



Abb. 2: Personen mit mehreren Adressen

Funktionskreise

Die Zuordnung von Personen zu einer Veranstaltung erfolgt u. a. mittels der Funktionskreise. Hierzu werden bei der tatsächlichen Veranstaltungsplanung die für die jeweilige Veranstaltung relevanten Funktionskreise festgelegt (vgl. Abschnitt „Veranstaltungsplanung und -durchführung“) und damit diejenigen Personen ausgewählt, die zu diesen Funktionskreisen gehören. Funktionskreise stellen also einen sehr wichtigen Mechanismus in WAVE dar.

Um die Funktionskreise übersichtlich verwalten zu können, ist ein mehrstufiger hierarchischer Funktionskreisbaum vorgesehen (Abb. 3). Dieser Funktionskreisbaum ist frei gestaltbar und nachträglich bearbeit- und erweiterbar.

Die Zuordnung von Personen zu Funktionskreisen wird in WAVE auf sehr komfortable Art und Weise unterstützt:

- Verschieben/Kopieren aller Personen aus einem Funktionskreis in einen anderen

- Verschieben/Kopieren ausgewählter Personen aus einem Funktionskreis in einen anderen
- Suchen von Personen ohne Funktionskreis
- Suchen von Personen ohne Adresse

- 1 Landesregierung NRW
 - 1.1 Ministerpräsident, Landesminister
 - 1.2 Staatssekretäre Landesministerien, Regierungssprecher
- 2 Parlamente
 - 2.1 Landtag NRW
 - 2.1.1 LT-Fraktionen
 - 2.1.1.1 LT-SPD-Fraktion

Abb. 3: Beispiel-Ausschnitt aus dem Funktionskreisbaum

Veranstaltungen

Wenn eine neue Veranstaltung geplant und durchgeführt werden soll, wird diese zunächst in WAVE erfasst. Für statistische Auswertungen ist das Klassifizieren von Veranstaltungen nach frei definierbaren Kategorien (z. B. kulturelle oder politische Veranstaltungen) möglich. Jedem Veranstaltungstyp kann ein Statistikennzeichen zugeordnet werden, ob dieser für die Statistik relevant oder nicht relevant ist.

Veranstaltungsplanung und -durchführung

Die Abbildung 4 zeigt, wie sich der zeitliche Ablauf einer Veranstaltung gestaltet:

Personen aus dem Personenstamm werden mittels der Funktionskreise zu einer Veranstaltung zugewiesen, so dass sehr schnell ein Vorschlag von Teilnehmerinnen und Teilnehmern entsteht (siehe Abb. 5).

Dieser Vorschlag kann weiter bearbeitet werden, indem einzelne Teilnehmer/-innen hinzugefügt oder entfernt werden. Hierbei wird die Benutzerin bzw. der Benutzer unterstützt durch eine komfortable Suchfunktion. Anhand beliebiger Suchkriterien können weitere Personen gefunden werden, z. B. Mitglieder/

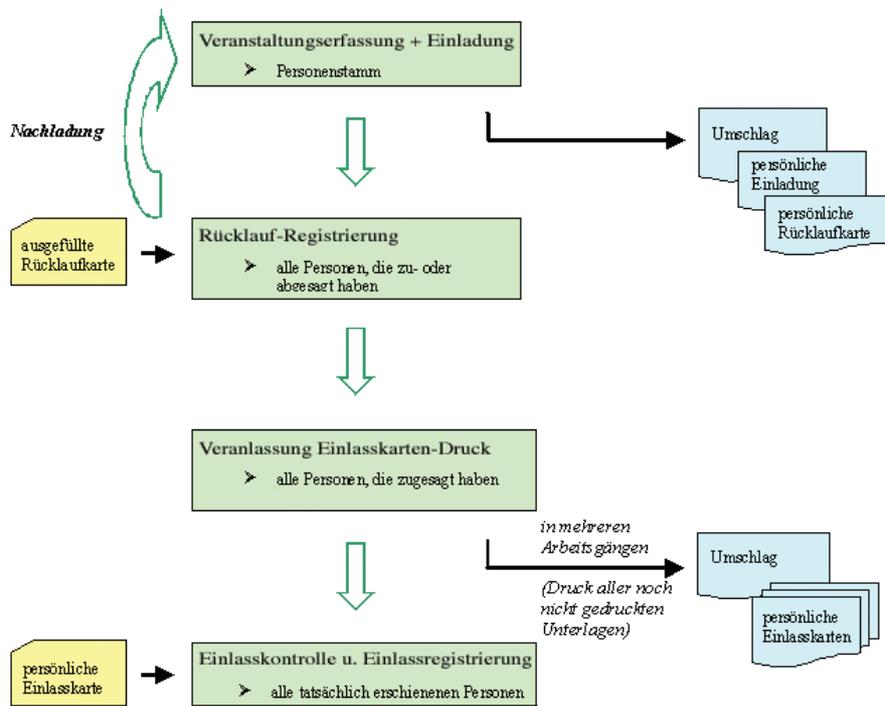


Abb. 4: Geschäftsprozess „Veranstaltungsplanung und -durchführung“

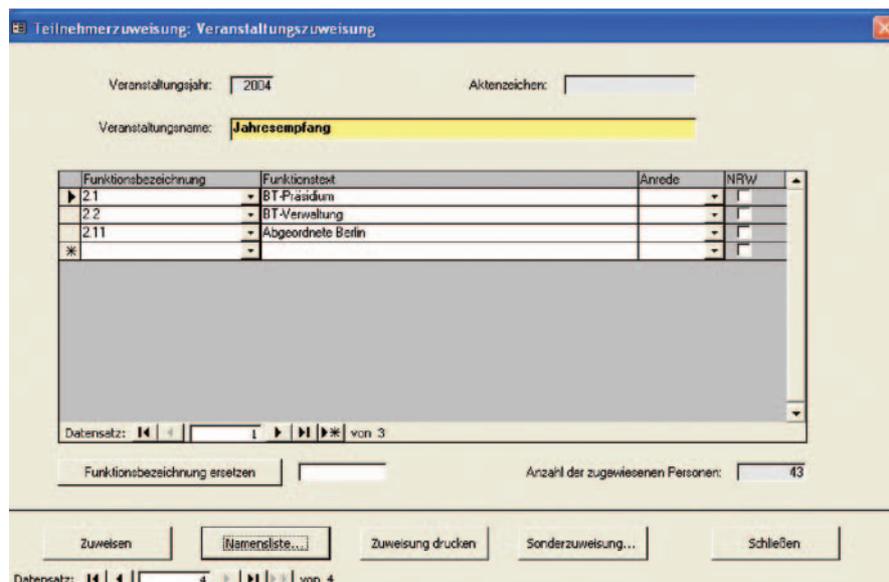


Abb. 5: Teilnehmerzuweisung – Teilnehmervorschlag erstellen

-innen eines anderen Funktionskreises, aber auch Anwohner/-innen einer bestimmten Straße (siehe Abb. 6).

Nachdem eine Treffermenge ermittelt wurde, können durch Markieren einzelne Personen oder auch alle gefundenen Personen in den Teilnehmervorschlag übernommen werden. Ein wesentlicher Punkt ist hierbei, dass WAVE sicherstellt, dass jede(r) Teilnehmer/-in im Teilnehmervorschlag nur einmal aufgeführt wird (auch bei mehrfacher Auswahl über ver-

schiedene Suchkriterien), um hierdurch doppelte Einladungen zu vermeiden.

Der so entstandene Teilnehmervorschlag wird in der sog. Namensliste festgehalten (siehe Abb. 7).

Ist der endgültige Teilnehmervorschlag erstellt, können die Teilnehmer/-innen durch persönlich adressierte Serienbriefe und Serien-E-Mails angeschrieben werden. Dabei werden persönliche Umschläge, Einladungskarten und

Abb. 6: Teilnehmerzuweisung – Suche

Name	Merkmal
Barthel, Eckhardt	Deutscher Bundestag
Benneter, Klaus-Uwe	Deutscher Bundestag
Butalikakis, Verena	Deutscher Bundestag
Dams, Klaus	Deutscher Bundestag
Dzembitzki, Dettel	Deutscher Bundestag
Eichtstädt-Bohlig, Franziska	Deutscher Bundestag
Gewall, Roland	Deutscher Bundestag
Groos, Hartmut	Deutscher Bundestag
Hadamek, Ruth	Deutscher Bundestag
Handke-Leptien, ...	Deutscher Bundestag
Hefas, Siegfried	Deutscher Bundestag
Kühl-Meyer, Beatrix	Deutscher Bundestag
Künast, Renate	Bundesministerium für ...
Leptien, Hans	Deutscher Bundestag
Linn, Susanne	Deutscher Bundestag
Löning, Markus	Deutscher Bundestag
Lötzsch, Gesine	Deutscher Bundestag

Abb. 7: Teilnehmerzuweisung – Namensliste und Suchergebnisse

Name	Vorname	Bemerkung	gedruckt	gekommt
Dümpke	Ekin	Hauptperson	✓	✓
Dümpe-Krüger	Dümpe-Krüger, Julia	Hauptperson	✓	✓
Kirstein	Donis	Begleitperson	✓	✓
Mika	Monika	Begleitperson	✓	✓
Sperling	Jürgen	Begleitperson	✓	✓
Eichhorn	Eichhorn, Maria	Hauptperson	✓	✓
Fischbach	Ingrid	Hauptperson	✓	✓
Gradistanac	Renate	Hauptperson	✓	✓
Graf	Angelika	Hauptperson	✓	✓
Haupt	Klaus	Hauptperson	✓	✓
Lenke	Ina	Hauptperson	✓	✓
Bätzing	Sabine	Hauptperson	✓	✓
Bätzing	Helmut	Begleitperson	✓	✓
Blumenthal	Antje	Hauptperson	✓	✓
Marks	Caren	Hauptperson	✓	✓
Kirstein	Donis	Sonstige Person	✓	✓
Kleinschmidt	Hans-Wolfgang	Sonstige Person	✓	✓
IT-Service	IT-Service	Sonstige Person	✓	✓

Abb. 8: Veranstaltungsdurchführung – Einlasskontrolle

Rücklaufkarten sowie Adresstiketten erzeugt.

Nachdem die ausgefüllte Rücklaufkarte wieder zurückgekehrt ist, wird nun festgehalten, wer zu- oder abgesagt hat. Ferner ist es möglich, hier die Begleitpersonen sowie sonstige Personen und deren Begleitung (also Personen außerhalb des Personenstamms) zu erfassen. Ggf. kann der einzuladende Personenkreis aus dem Personenstamm auch noch erweitert und somit zusätzliche Personen nachgeladen werden.

Da die Rücklaufkarten nicht zu genau einem Zeitpunkt, sondern zeitversetzt eintreffen, kann der Druck der entsprechenden Unterlagen (persönliche Einlasskarten, Umschläge, Begleitzettel, ...) nach Eingang veranlasst werden. D. h. für alle Personen, die zwischenzeitlich ihre Teilnahme an der Veranstaltung mitgeteilt haben und die noch keine Unterlagen erhalten haben, werden die Einlasskarten, Umschläge und ggf. weitere Dokumente gedruckt.

Am Tag der eigentlichen Veranstaltung erfolgt nach entsprechender Einlasskontrolle die Registrierung der tatsächlich erschienenen Personen. Die Einlasskontrolle bezieht sich auf alle zugesagten Personen inklusive der Begleitpersonen und sonstiger Personen (siehe Abb.8). Mittels einer Suchfunktion können die erschienenen Personen gesucht und gekennzeichnet werden. Hier ist auch der Einsatz eines Barcode-Scanners denkbar, mit dem die Person anhand ihrer Einlasskarte sehr schnell gefunden werden kann. Ist ein Bild der gefundenen Person vorhanden, kann dieses angezeigt werden, um so die Personenidentität zu überprüfen – das Vorzeigen eines Personalausweises ist in diesen Fällen nicht mehr erforderlich.

Serienbriefe/Serien-E-Mails/ Etiketten/Listen

WAVE bietet die Möglichkeit, über eine veranstaltungsbezogene Schnittstelle zu Word, persönliche Serienbriefe zu erzeugen. Im System sind mehre-

re Mustervorlagen enthalten, wobei individuelle Vorlagen ebenfalls erstellbar sind. Nach Auswahl einer Serienbriefvorlage wird das entsprechende Word-Dokument mit den entsprechenden Teilnehmer(inne)n (diejenigen die per Brief eingeladen werden sollten) erzeugt und in einem frei zu wählenden Ablage-Verzeichnis abgelegt. Ebenfalls möglich ist der Versand persönlich adressierter Serien-E-Mails, der auch Anlagen beigelegt werden können.

Darüber hinaus bietet WAVE die Möglichkeit zum Drucken von veranstaltungsunabhängigen Listen (z. B. Namensliste der Mitglieder/-innen eines Funktionskreise) und von Etiketten.

Statistik

Nach Abschluss einer Veranstaltung wird diese in die Statistik überführt. Die Statistik über eine durchgeführte Veranstaltung enthält die Anzahl der Einladungen, Anzahl der Absagen und Anzahl der tatsächlichen Teilnehmer/-innen.

Systemumgebung

WAVE ist eine Access2003-Anwendung, die als Datenbanksystem wahlweise die Access eigene Datenbank (die Jet Engine) oder SQL-Server 2000 verwendet und ein Windows Betriebssystem benötigt.

Leistungsmerkmale

WAVE ist in einigen Punkten bisher speziell auf die Erfordernisse der Landesvertretung Nordrhein-Westfalen zugeschnitten, jedoch im Wesentlichen universell einsetzbar. Das System zeichnet sich dadurch aus, dass es die Planung und Durchführung von Großveranstaltungen deutlich erleichtert.

Zusammenfassend lassen sich die hauptsächlichsten Leistungsmerkmale wie folgt beschreiben:

- Zeitersparnis bei der Planung großer Veranstaltungen

- Unterstützung bei der gezielten Ermittlung von Teilnehmer(inne)n
- Vermeidung doppelter Einladungen
- Rücklauf- und Einlasskontrolle
- Übernahme von Kontakten aus MS Outlook
- Papierkorb-Funktion
- Gruppenbezogene Zugriffsrechte für die Personendaten
- Zentrale Datenhaltung

Nähere Informationen erhalten Sie bei der Autorin und den Autoren.

Monika Mika
Telefon: 0211 9449-6739
E-Mail: monika.mika@lds.nrw.de

Jürgen Sperling
Telefon: 0211 9449-6729
E-Mail: juergen.sperling@lds.nrw.de

Dipl.-Inform. Ulrich von Hagen
Telefon: 0211 9449-6706
E-Mail: ulrich.von-hagen@lds.nrw.de

rz lds
Rechenzentrum im LDS NRW.

Ihre Vorteile

Wir helfen Ihnen, die „richtige“ Archivierung Ihrer Daten zu finden und zu verwirklichen.

Unsere Stärken sind Ihre Vorteile:

- analoge Archivierung seit 35 Jahren
- digitale Archivierung seit 10 Jahren
- selbsttragende Archive auf CD/DVD
- zentrale Archive Online
- Datenverarbeitung und Datenhaltung im Sicherheitsbereich
- Komplettlösungen vom Papier oder von der Datei hin zum digitalen Archiv
- Beratung zu Langzeitstrategien

Wir freuen uns auf Ihre Aufgabenstellung und unterbreiten Ihnen gern unser Angebot.

Kontakt

Archivierung
Telefon: 0211 9449-2180
E-Mail: rechenzentrum@lds.nrw.de

Weitere Informationen finden Sie unter:
<http://lv.landesintranet.lds.nrw.de/datenverarbeitung/rechenzentrum/archivier/index.html>

Archivierung. Eine Dienstleistung des Landesamtes für Datenverarbeitung und Statistik Nordrhein-Westfalen. 2005

© LDS NRW, Düsseldorf, 2005

LDS NRW.

Das neue V-Modell XT



Das V-Modell 97 (auch bekannt unter dem offiziellen Namen „Entwicklungsstandard für IT-Systeme des Bundes“ (EStdIT)¹⁾ ist für viele Unternehmen und Behörden in den letzten Jahren eine entscheidende Grundlage für die Organisation und Durchführung von IT-Vorhaben geworden. Es verbessert zum einen die Qualität der erstellten Systeme und unterstützt zum anderen die Zusammenarbeit zwischen Auftraggebern und Auftragnehmern. Für die Landesverwaltung Nordrhein-Westfalen wurde in den IT-Richtlinien NRW festgelegt, dass bei der Durchführung von IT-Vorhaben das V-Modell zugrunde gelegt werden soll. Das Vorgehensmodell des LDS NRW (ABS = Anwendungsbereitstellungsstandard) beruht auf dem V-Modell 97.

Nach der Fertigstellung des V-Modells im Jahre 1997 hat es allerdings keine Fortschreibung gegeben, so dass das V-Modell nicht in allen Bereichen den aktuellen Stand der Informationstechnologie widerspiegelt.

Zielsetzung des V-Modell XT

Seit dem Jahre 2002 erfolgte im V-Modell-Weiterentwicklungsprojekt WEIT²⁾ eine grundlegendere Überarbeitung des V-Modells. Beteiligt waren hieran die Technischen Universitäten München und Kaiserslautern sowie als Industriepartner die Firmen 4Soft GmbH, EADS, IABG mbH und Siemens AG.

Mit der Weiterentwicklung des V-Modells wurden dabei insbesondere folgende Ziele verfolgt:

- bessere Unterstützung der Anpassbarkeit, Anwendbarkeit, Skalierbarkeit sowie Änder- und Erweiterbarkeit des V-Modells
- Erweiterung des Anwendungsbereiches und Ausweitung auf den gesamten Systemlebenszyklus
- Berücksichtigung des neuesten Stands der Technologie
- Anpassung an aktuelle Vorschriften und Normen
- Einführung eines organisationspezifischen Anpassungs- und Verbesserungsprozesses für Vorgehensmodelle

1) V-Modell® ist eine eingetragene Marke der Bundesrepublik Deutschland.
2) Das Projekt WEIT („Weiterentwicklung des Entwicklungsstandards für IT-Systeme des Bundes auf Basis des V-Modell 97“) wird im Auftrag der Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung (KBSt) und des Bundesamts für Informationsmanagement und Informationstechnik der Bundeswehr (IT-AmtBw) durchgeführt.
3) Der Namenszusatz „XT“ zu V-Modell steht für „extreme tailoring“, oder aber für „extendable“.

Das neue V-Modell XT³⁾ unterstützt eine ziel- und ergebnisorientierte Vorgehensweise. Hierzu trägt u. a. bei, dass stärker als im V-Modell 97 die Produkte im Mittelpunkt stehen. Der Projektfortschritt orientiert sich also an den jeweils zu erstellenden Produkten (während beim V-Modell 97 stärker die Aktivitäten im Vordergrund standen). Für jedes Produkt ist dabei eindeutig eine bestimmte Rolle verantwortlich und damit im Projekt eine dieser Rolle zugeordnete Person. Durch definierte Vorgaben für die Produkte sowie durch explizite Produktabhängigkeiten wird die Produktqualität überprüfbar.

Beispiele für Produktabhängigkeiten.

- Die in der Anwenderaufgabenanalyse ermittelten Anwenderaufgaben, Anwenderprofile und die physische Benutzungsumgebung sind als Input für das Thema „Funktionale Anforderungen“ in der Gesamtsystemspezifikation (Pflichtenheft) zu berücksichtigen.
- Die vertraglich vereinbarte Mitwirkung des Auftraggebers bei Aktivitäten des Auftragnehmers muss im Projektplan festgelegt werden.
- Das QS-Handbuch enthält Vorgaben zur Prüfung der Systemelemente, die in den Implementierungs-, Integrations- und Prüfkonzepten berücksichtigt werden müssen.

Struktur und Aufbau des V-Modell XT

Anstelle der aus dem V-Modell 97 bekannten vier Submodelle „Systemerstellung“, „Projektmanagement“, „Qualitätssicherung“ und „Konfigurationsmanagement“ gibt es im V-Modell XT nun 19 sog. Vorgehensbausteine, wobei diese Vorgehensbausteine deutlich feiner geschnitten sind als die bisherigen Submodelle. Diese Vorgehensbausteine bilden die modularen Einheiten des V-Modell XT. Ein Vorgehensbaustein kapselt Produkte, Rollen und Aktivitäten (siehe Abb. 1). Vorgehensbausteine können eigenständig verwendet werden. Mit Hilfe der Vorgehensbausteine kann ein organisations- bzw. projektspezifisches Vorgehensmodell erstellt werden. Zudem sind sie die Einheit, die bei der Fortschreibung des V-Modells unabhängig verändert und weiterentwickelt werden kann.

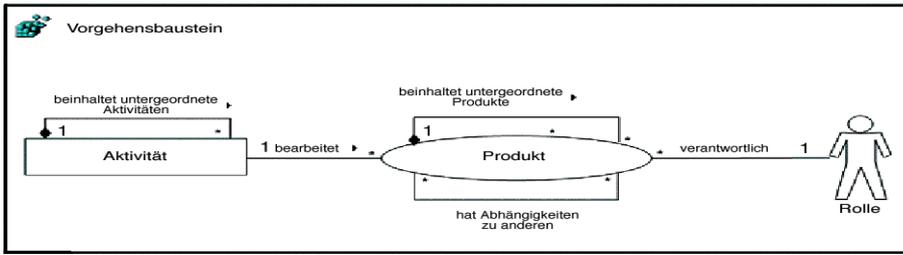


Abb. 1: Vorgehensbaustein⁴⁾

Vorgehensbausteine, Produkte und Aktivitäten machen keine Vorgaben hinsichtlich der Durchführungsreihenfolge. Hier kommen nun die Projektdurchführungsstrategien ins Spiel.

Eine Projektdurchführungsstrategie legt die Reihenfolge der zu erreichenden Projektfortschrittsstufen fest. Das V-Modell unterstützt verschiedene Projektdurchführungsstrategien (siehe Abschnitt „V-Modell XT für unterschiedliche Projekttypen“). Entscheidungspunkte wiederum am Ende einer bestimmten Projektfortschrittsstufe fertig gestellt sein muss.

Einen Überblick über das Zusammenspiel der einzelnen Elemente des V-Modell XT gibt Abb. 2.

hensbausteinen zusammen. Hierbei gibt es Vorgehensbausteine, die nur den Auftraggeber betreffen (z. B. den Vorgehensbaustein „Anforderungsfestlegung“) und solche, die nur den Auftragnehmer betreffen (z. B. „SW-Entwicklung“) und es gibt auch Vorgehensbausteine, die sowohl für den Auftraggeber als auch für den Auftragnehmer relevant sind (z. B. „Systemsicherheit“). Auf die Schnittstellen zwischen Auftraggeber- und Auftragnehmer-Projekten wird im folgenden Abschnitt näher eingegangen.

Darüber hinaus gibt es auch einen Vorgehensbaustein, der sich auf die Entwicklung eines organisationspezifischen Vorgehensmodells bezieht.

grundlegende Managementmechanismen, die stets anzuwenden sind. Einen Überblick über alle Vorgehensbausteine des V-Modell XT gibt Abb. 3.

Die in der Abb. 3 durch Pfeile dargestellten Vorgehensbausteinabhängigkeiten bewirken, dass die Auswahl eines Vorgehensbausteins die Auswahl anderer Vorgehensbausteine nach sich zieht.

Es gibt dabei zwei unterschiedliche Vorgehensbausteinabhängigkeiten:

- Die verpflichtende Vorgehensbausteinabhängigkeit: Die Auswahl von Vorgehensbaustein A zieht direkt die Auswahl von Vorgehensbaustein B nach sich.
- Die alternative Vorgehensbausteinabhängigkeit: Wenn Vorgehensbaustein A gewählt wird, muss mindestens einer der Vorgehensbausteine B bis X gewählt werden.

V-Modell XT für unterschiedliche Projekttypen

Das V-Modell XT unterstützt unterschiedliche Projekttypen:

- Systementwicklungsprojekt eines Auftraggebers
- Systementwicklungsprojekt eines Auftragnehmers
- Entwicklung eines organisationspezifischen Vorgehensmodells

Für ein Systementwicklungsprojekt eines Auftraggebers kommt dabei die Projektdurchführungsstrategie „Vergabe und Durchführung von Systementwicklungsprojekten“ zum Tragen (vgl. Abb. 4). Die einzelnen Kästchen stellen hierbei Entscheidungspunkte dar.

Für Systementwicklungsprojekte eines Auftragnehmers sieht der Ablauf grundsätzlich wie in Abb. 5 dargestellt aus.

Die Entwicklungsprojekte eines Auftraggebers und eines Auftragnehmers laufen aber selbstverständlich nicht losgelöst voneinander sondern haben

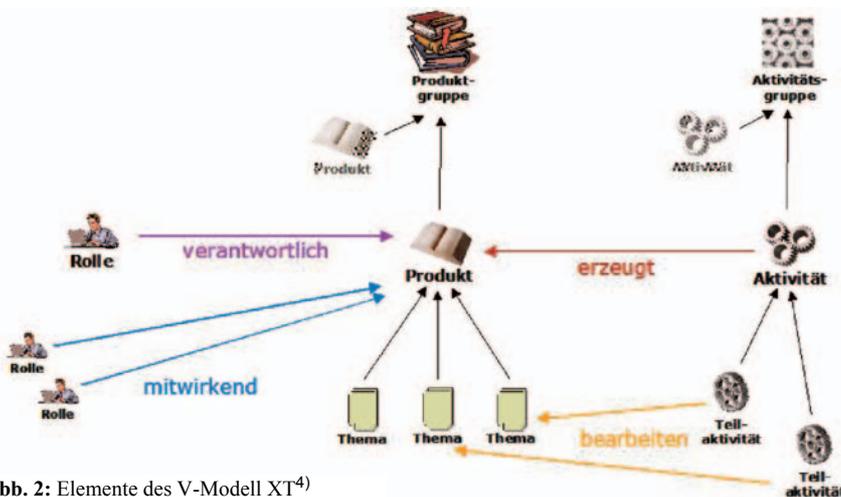


Abb. 2: Elemente des V-Modell XT⁴⁾

Vorgehensbausteine des V-Modell XT

Wie bereits oben erwähnt, setzt sich das V-Modell aus einzelnen Vorge-

Der V-Modell-Kern umfasst diejenigen Vorgehensbausteine, die auf jeden Fall – also unabhängig vom jeweiligen Projekttyp – zu verwenden sind; dies sind die Vorgehensbausteine „Projektmanagement“, „Qualitätssicherung“, „Problem- und Änderungsmanagement“ und „Konfigurationsmanagement“ (in Abb. 3 der Bereich mit der dickeren Umrandung). Der V-Modell-Kern definiert also

4) Die verwendete Grafik wurde freundlicherweise von Herrn J. Prof. Dr. Rausch (TU Kaiserslautern, WEIT-Projektleiter) bzw. vom WEIT-Team zur Verfügung gestellt.

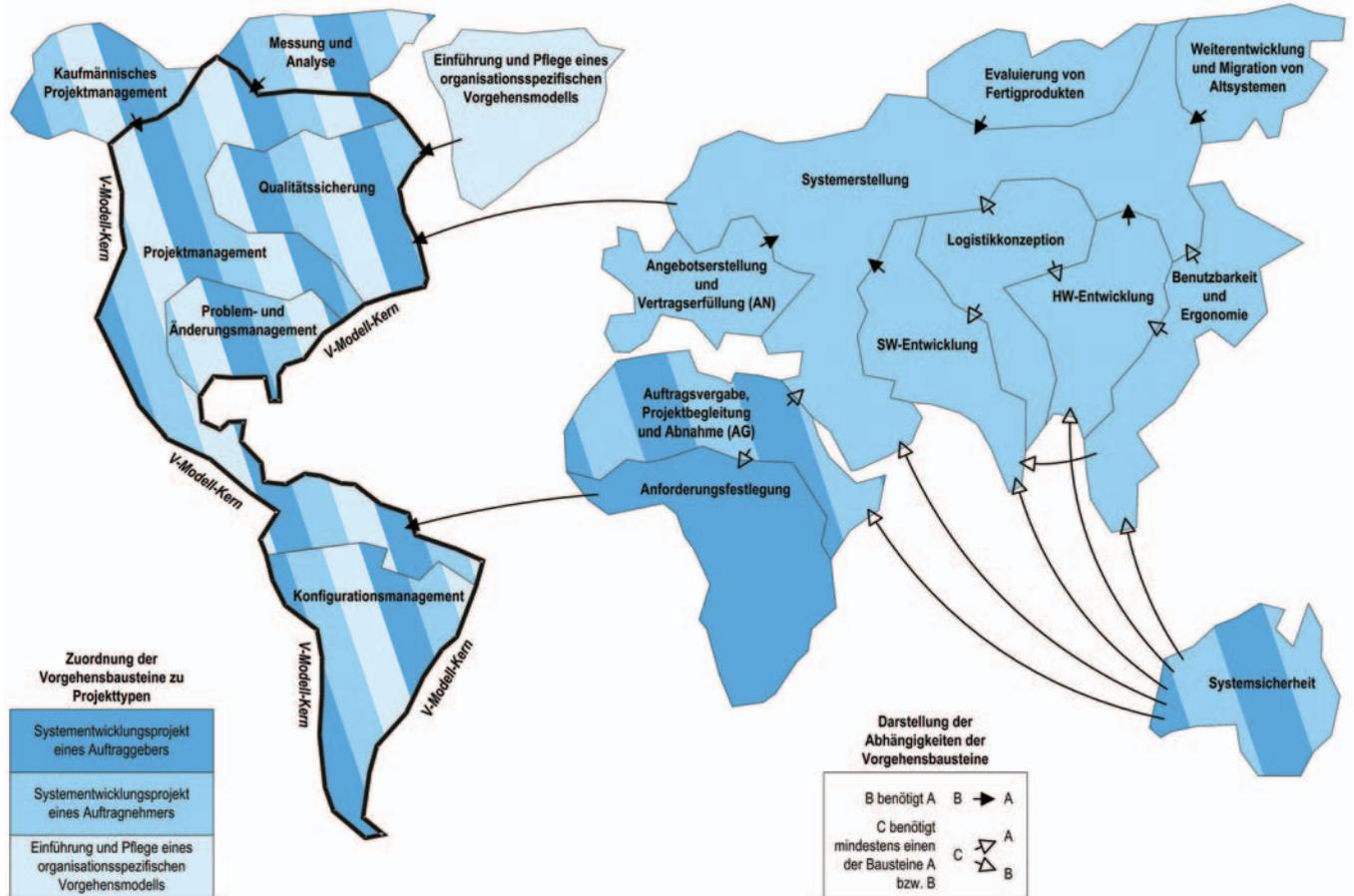


Abb. 3: Übersicht über die Vorgehensbausteine des V-Modell XT (Vorgehensbausteinlandkarte)⁴⁾

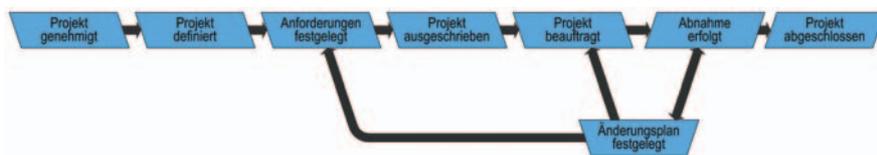


Abb. 4: Systementwicklungsprojekt eines Auftraggebers⁴⁾

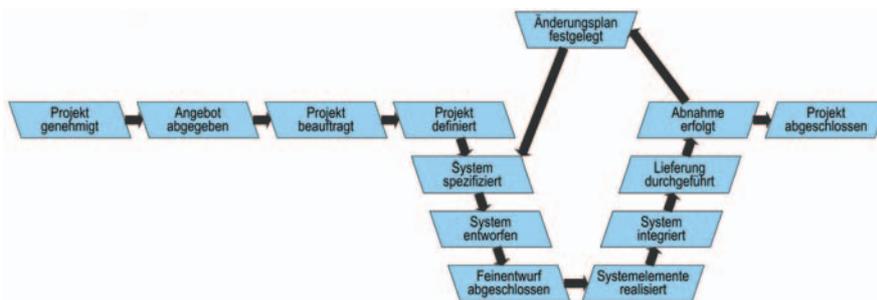


Abb. 5: Systementwicklungsprojekt eines Auftragnehmers⁴⁾

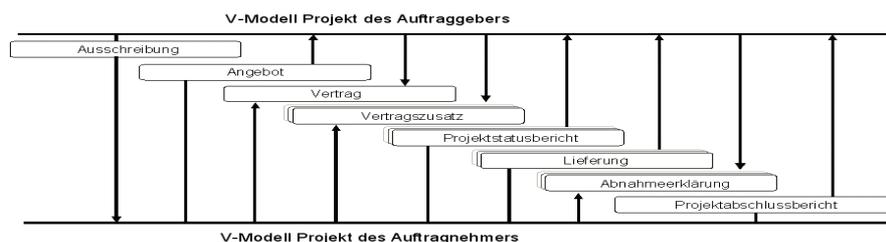


Abb. 6: Schnittstelle zwischen Auftraggeber und Auftragnehmer⁴⁾

untereinander verschiedene Schnittstellen (vgl. Abb. 6).

In das V-Modell XT wurden auch Mechanismen aufgenommen, um das Vorgehensmodell selbst in eine Organisation einzuführen und dabei an die organisationspezifischen Gegebenheiten anzupassen und entsprechend der Bedürfnisse der jeweiligen Organisation fortzuentwickeln.

Hier kommt die Projektdurchführungsstrategie „Einführung und Pflege eines organisationspezifischen Vorgehensmodells“ zum Tragen. Das Projektziel eines hierauf beruhenden V-Modell-Projekts ist es also, in einer Organisation ein Vorgehensmodell zu etablieren und dieses kontinuierlich zu verbessern.

Einen Überblick über alle im V-Modell XT vorgesehenen Projektdurchführungsstrategien gibt Abb. 7. Die Abbildung zeigt dabei die derzeit verfügbaren Projektdurchführungsstrategien in

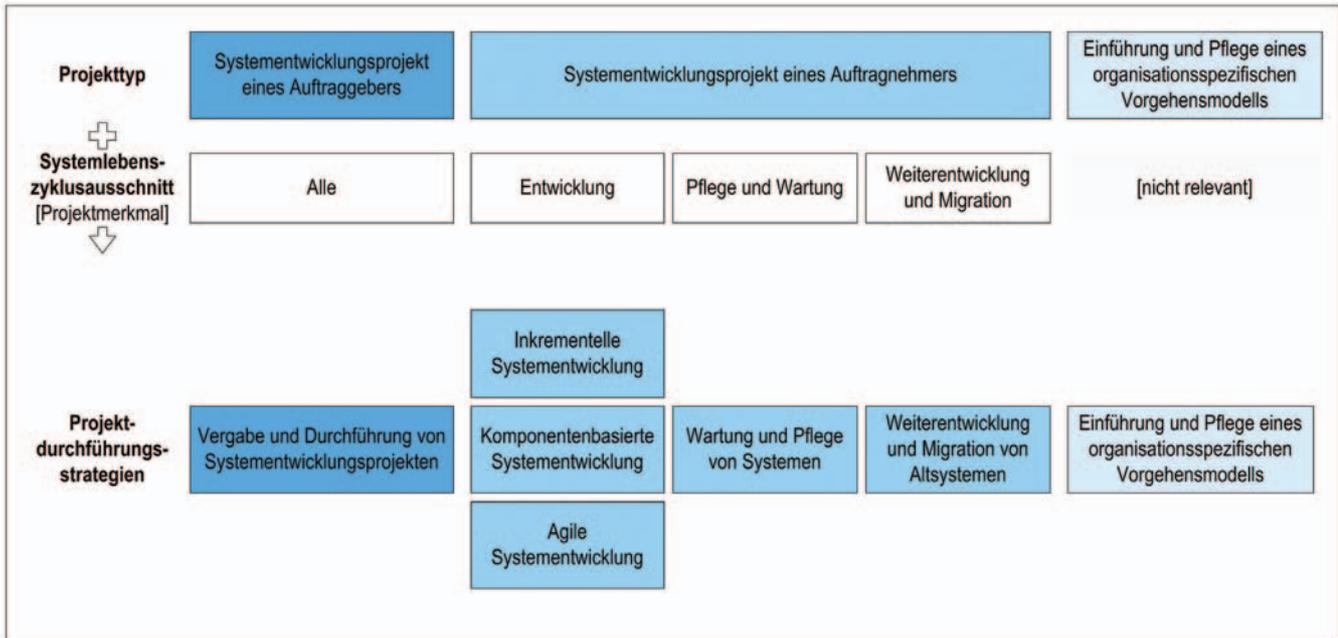


Abb. 7: Zuordnung der Projektdurchführungsstrategien zu den Projekttypen⁴⁾

Abhängigkeit vom Projekttyp und vom Systemlebenszyklusausschnitt.

Anpassung des V-Modell XT an Organisationen und Projektsituationen

Da das V-Modell XT – als generisches Vorgehensmodell – für ein weites Spektrum unterschiedlicher Projekte verwendbar ist, muss das Vorgehensmodell vor der eigentlichen Nutzung in einem Projekt an die jeweiligen Projektgegebenheiten angepasst werden.

Dieser unter dem Namen „Tailoring“ bekannte Anpassungsprozess erfolgt beim

V-Modell XT über die Auswahl der im Projekt anzuwendenden Vorgehensbausteine. Hierbei sind die Vorgehensbausteine des VM-Kerns stets auszuwählen, d. h. sie stellen die Mindestmenge der Vorgehensbausteine dar. Im Gegensatz zum Tailoring mit Streichbedingungen beim V-Modell 97, bei dem ausgehend vom gesamten V-Modell einzelne Aktivitäten herausgeschnitten wurden, verfolgt das Tailoring beim V-Modell XT einen anderen Ansatz: hier werden – ausgehend vom VM-Kern – soweit erforderlich weitere Vorgehensbausteine hinzugenommen. Der Projekttyp und die Projektdurchführungsstrategie mit ihren dazugehörigen Entscheidungspunkten wird ebenfalls beim Tailoring ermittelt.

Und so werden die jeweiligen Projektgegebenheiten beim Tailoring berücksichtigt:

Zunächst wird durch die Bestimmung der Projektmerkmale ein Anwendungsprofil erstellt (vgl. Abb. 8). Dieses Anwendungsprofil legt dabei u. a. den Projekttyp fest. Der Projekttyp führt zusammen mit den Projektmerkmalen zu einer Auswahl der verpflichtend und optional zu verwendenden Vorgehensbausteine. Ebenso grenzen der Projekttyp und das Anwendungsprofil bereits die anwendbaren Projektdurchführungsstrategien und die damit verbundenen zu durchlaufenden Entscheidungspunkte ein.

Projektmerkmale	Anwendungsprofil
Projektgegenstand	SW-System
Projektrolle	Auftraggeber
Systemlebenszyklus-Ausschnitt	Entwicklung
Wirtschaftliches Projektmanagement	Entwicklung Pflege und Wartung Weiterentwicklung und Migration
Quantitative Projektkennzahlen	
Fertigprodukte	Ja
Benutzerschnittstelle	Ja
Safety und Security	Nein
Hohe technologische Risiken	Nein

Mit Hilfe der **Projektmerkmale** lässt sich ein konkretes Projekt charakterisieren. Jedes Projektmerkmal wird hierzu mit einem seiner möglichen Werte belegt. Hierdurch entsteht ein sog. **Anwendungsprofil**.

Das Anwendungsprofil ist also eine Charakterisierung des Projektes in Bezug auf die vom V-Modell vorgegebenen Projektmerkmale. Für jedes dieser Projektmerkmale sind feste Werte vorgegeben, aus denen einer auszuwählen ist.

Abb. 8: Durchführung des Tailorings: Auswahl eines Anwendungsprofils⁴⁾

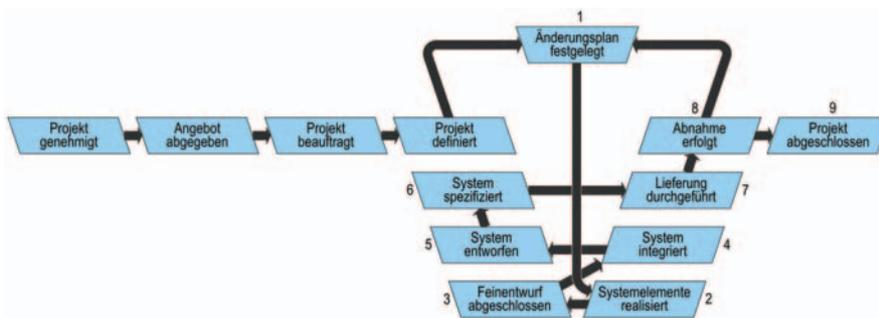


Abb. 9: Projektdurchführungsstrategie „Agile Systementwicklung“⁽⁴⁾

Handelt es sich z. B. um ein Systementwicklungsprojekt eines Auftragnehmers, so sieht das V-Modell XT für den Systemlebenszyklusabschnitt „Entwicklung“ drei verschiedene Projektdurchführungsstrategien vor:

- inkrementelle Systementwicklung
- komponentenbasierte Systementwicklung
- agile Systementwicklung (vgl. Abb. 9)

Wie man sieht, wurden beim V-Modell XT konsequent neuere Entwicklungen auf dem Gebiet der Vorgehensweisen bei der Systementwicklung aufgegriffen (zu den agilen Vorgehensweisen vgl. auch den Beitrag „Agil und extrem – neue Wege in der Softwareentwicklung“ in der Ausgabe 2/2004 der LDVZ-Nachrichten).

Auf eine Projektdurchführungsstrategie für ein starres Wasserfallvorgehen wurde im V-Modell XT verzichtet.

Welche der oben genannten drei Projektdurchführungsstrategien nun für ein Entwicklungsprojekt vom Auftragnehmer adäquat ist, hängt vom jeweiligen Anwendungsprofil ab, d. h. es wird die Projektdurchführungsstrategie ausgewählt, die am besten zu den Projektgegebenheiten passt.

Die Agile Systementwicklung z. B. wird in der V-Modell-Dokumentation wie folgt charakterisiert:

„Die Agile Systementwicklung (AN) basiert auf der Erkenntnis, dass es oft nicht möglich ist, die Anforderungen an ein System vorab zu definieren. Außerdem stellt sie sicher, dass nichts spezifiziert wird, was sich als nicht re-

alisierbar herausstellt. Somit wird diese Strategie insbesondere verwendet, wenn hohe Realisierungsrisiken im Projekt vorhanden sind. Änderungen an den Anforderungen werden über das Problem- und Änderungsmanagement verwaltet, im Entscheidungspunkt 'Änderungsplan festgelegt' wird über die Annahme dieser Änderungen entschieden und sie werden im Rahmen der Systementwicklung auch regelmäßig eingeplant. Typisch für die Agile Systementwicklung (AN) ist darüber hinaus die Präsenz des Auftraggebers auf der Auftragnehmerseite während der Entwicklung. Dadurch kann der Auftraggeber Änderungswünsche sehr direkt übermitteln. Der Auftragnehmer entwirft, realisiert und liefert das System dann, ähnlich wie bei der Projektdurchführungsstrategie Inkrementelle Systementwicklung (AN), in einzelnen Stufen. Diese Stufen werden jede für sich vom Auftraggeber abgenommen. Für den Auftraggeber hat diese Vorgehensweise den Vorteil, dass er bereits frühzeitig in den Besitz eines lauffähigen Systems gelangt, das die wichtigsten Grundfunktionalitäten realisiert. Ferner ermöglicht sie eine frühzeitige Rückmeldung durch den Auftraggeber, die die Entwicklungsrisiken des Auftragnehmers minimiert.“

Werkzeuge zum neuen V-Modell

Zur Unterstützung des V-Modell XT werden zwei Open Source-Werkzeuge zur Verfügung gestellt:

- V-Modell-XT-Editor
- V-Modell-XT-Projektassistent

Der V-Modell-XT-Editor kann dazu genutzt werden, das durch das V-Modell XT vorgegebene Rahmenwerk an branchen- oder unternehmensspezifische Bedürfnisse anzupassen. Mit ihm können neue Vorgehensbausteine hinzugefügt oder die Struktur der Projektvorlagen ergänzt werden.

Der V-Modell-XT-Editor wurde auch bereits bei der Entwicklung des V-Modell XT verwendet. Es handelt sich hierbei um einen formularbasierten, strukturierten Editor, der als Plugin der Entwicklungsplattform Eclipse realisiert wurde. Der Editor sowie das zugrunde liegende Framework 4Ever der 4Soft GmbH wurden im Rahmen des vom Bundesministerium für Bildung und Forschung geförderten Projekts NOW als Open Source Software verfügbar gemacht und werden seither kontinuierlich weiterentwickelt.

Der V-Modell-XT-Projektassistent stellt die Referenzimplementierung für den im V-Modell XT spezifizierten Tailoring-Mechanismus dar. Mit seiner Hilfe können für ein konkretes Projekt die Vorgaben des V-Modell XT an die entsprechende Projektsituation angepasst werden. Der Projektassistent ermöglicht die Auswahl von Projektmerkmalen, die konsistente Kombination von Vorgehensbausteinen und die Festlegung von Projektdurchführungsstrategien. Das auf diese Weise angepasste V-Modell XT kann anschließend als HTML-Seiten exportiert werden. Für die nachfolgende Version des Projektassistenten sind u. a. zusätzliche Export-Formate sowie die Generierung von Produktvorlagen geplant.

Darüber hinaus sind auch bereits einige kommerzielle Werkzeuge, die das V-Modell XT unterstützen, verfügbar bzw. in der Entwicklung. Zu nennen ist hier insbesondere die in-Step V-Modell-XT-Edition der Projektmanagement-Software in-Step der Fa. microTOOL.

V-Modell im Internet

Seit Anfang November 2004 ist das neue V-Modell XT im Internet verfügbar.

Die Informationen zum V-Modell XT wurden in den Internetauftritt der KBSt (Koordinierungs- und Beratungsstelle der Bundesregierung für Informationstechnik in der Bundesverwaltung) integriert.

Bei der Beschreibung des V-Modells wurde versucht, für die verschiedenen Projektbeteiligten unterschiedliche Zugänge zu den jeweils relevanten Informationen zu schaffen. Aufbauend auf einer Darstellung der Grundlagen des V-Modells und einer Tour durch das V-Modell, werden mittels sog. V-Modell-Referenzen unterschiedliche Sichten auf das V-Modell XT angeboten (vgl. auch Abb. 10).

Die V-Modell-XT-Dokumentation wird als PDF-Download angeboten. Die Teile des V-Modells können dabei entweder einzeln oder als Gesamtdokument heruntergeladen werden.

Daneben wird eine Online Version des V-Modells angeboten, die zusätzlich auch als komprimiertes Archiv heruntergeladen werden kann.

Die Produktvorlagen stehen als Word-Templates zur Verfügung. Das V-Modell bietet dabei jeweils Vorlagen aller während der Projektplanung und -durchführung zu erstellenden Produkte für die Projekttypen „Systementwicklungsprojekt eines Auftraggebers“, „Systementwicklungsprojekt eines Auftragnehmers“ und „Entwicklung eines organisations-spezifischen Vorgehensmodells“ an.

Zukünftig soll auch eine englische Version des V-Modell XT verfügbar gemacht werden.

Außerdem existiert noch eine Lerntour zum V-Modell XT. Ein Forum zum Erfahrungsaustausch für V-Modell-XT-Anwender wurde ebenfalls eingerichtet.

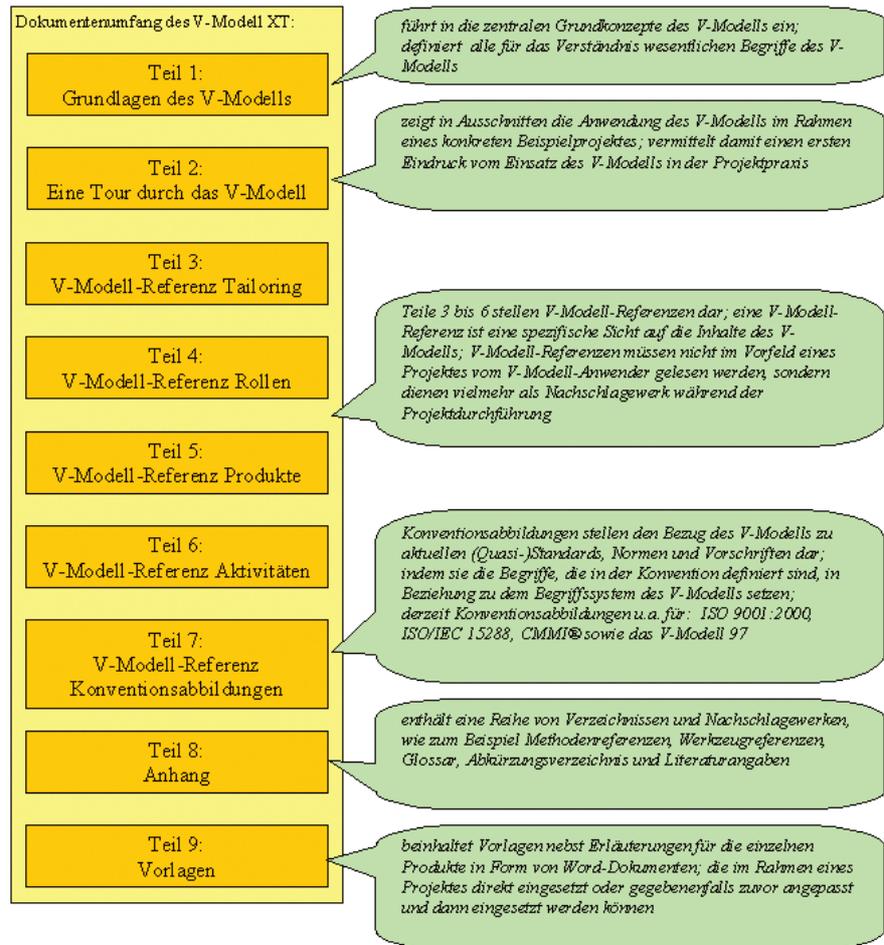


Abb. 10: Dokumentenumfang des V-Modell

Verbreitung des V-Modell XT

Das V-Modell XT ist als Komponente im sog. Methodenbaukasten für die Bundesverwaltung eingebettet (vgl. Abb. 11) und steht als Open Source-Produkt zur Verfügung.

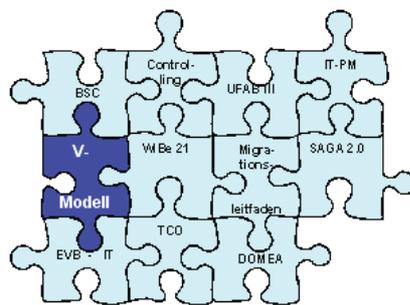


Abb. 11: Methodenbaukasten für die Bundesverwaltung⁵⁾

Der Interministerielle Koordinierungsausschuss für die Informationstechnik

⁵⁾ Die verwendete Grafik wurde freundlicherweise von Herrn Dr. Bliß (BMI-KBSt, Auftraggeber WEIT-Projekt) zur Verfügung gestellt.

in der Bundesverwaltung (IMKA) hat am 4. 11. 2004 das V-Modell XT als verbindlich für Neuentwicklungen in der Bundesverwaltung erklärt. Die Bundesverwaltung strebt auch an, dass sich das V-Modell XT als führender europäischer Standard für Vorgehensmodelle durchsetzt.

Am 4. Februar 2005 wurde das V-Modell XT an der Technischen Universität München in einer feierlichen Veranstaltung der Öffentlichkeit vorgestellt. Redner waren dabei unter anderem der Bundesminister des Innern Otto Schily und der Präsident des Bundesamtes für Informationsmanagement und Informationstechnik der Bundeswehr (IT-AmtBw) Wolfgang Stolp.

Zusätzliche Informationen zur Weiterentwicklung des V-Modells erhalten Sie beim Autor des Artikels, der sowohl als Mitglied der V-Modell-Änderungskonferenz als auch als Reviewer

beim V-Modell-Weiterentwicklungsprojekt aktiv an der Weiterentwicklung dieses Vorgehensmodells mitgewirkt hat.

Glossar zum V-Modell XT

Die nachfolgenden Begriffserklärungen wurden – mit kleineren redaktionellen Überarbeitungen – dem V-Modell XT entnommen.

Aktivität

Ein Aktivität (im Sinne eines Aktivitätstyps) beschreibt die konkreten Aktivitätsexemplare, die während eines Entwicklungsprozesses ausgeführt werden können.

Aktivitäten sind Bestandteil genau einer → Aktivitätsgruppe und damit stets einem → Vorgehensbaustein zugeordnet. Jedes → Produkt wird einer Aktivität zugeordnet, die es bearbeitet. Aktivitäten verändern also Produkte. Produkte, die in einer Aktivität nur als Eingabe dienen, werden nicht explizit einer Aktivität zugeordnet. Bei Fertigstellung eines Produkts ist dieses im → Bearbeitungszustand „fertig gestellt“ und die dem Produkt zugeordneten Fertigstellungsbedingungen gelten. Aktivitäten untergliedern sich weiter in → Teilaktivitäten.

Aktivitätsgruppe

Eine Aktivitätsgruppe ist stets genau einem → Vorgehensbaustein zugeordnet. Sie ist eine vorgehentechnisch motivierte, inhaltliche Gruppierung von → Aktivitäten. Eine Aktivitätsgruppe für sich genommen bearbeitet also keine → Produkte oder → Themen.

Anwendungsprofil

Ein Anwendungsprofil stellt die Wertbelegung der einzelnen → Projektmerkmale im konkreten → Projekt dar. Anhand dieses Anwendungsprofils findet ein erstes → Tailoring statt.

Auftraggeber

Unter einem Auftraggeber wird der Kunde im Rahmen einer Vertragssituation verstanden, also der Empfänger eines vom → Auftragnehmer bereitgestellten → Produkts (DIN EN ISO 8402).

Auftragnehmer

Unter einem Auftragnehmer wird der Lieferant im Rahmen einer Vertragssituation verstanden, also die Organisation, die dem → Auftraggeber ein → Produkt bereitstellt (DIN EN ISO 8402).

Auftraggeber-/Auftragnehmer-Schnittstelle

Die Auftraggeber-/Auftragnehmer-Schnittstelle beschreibt explizit, welche → Produkte zwischen dem Auftraggeber- und dem Auftragnehmer-V-Modell-Projekt ausgetauscht werden. Diese Produkte werden → Schnittstellenprodukte genannt.

Bearbeitungszustand

→ Produkte besitzen einen Bearbeitungszustand, der durch → Aktivitäten verändert werden kann. Man unterscheidet zwischen den drei Bearbeitungszuständen „in Bearbeitung“, „vorgelegt“ und „fertig gestellt“.

Entscheidungspunkt

In einem Entscheidungspunkt wird über das Erreichen einer → Projektfortschrittsstufe entschieden. Diese Entscheidung wird auf Basis der im Entscheidungspunkt festgelegten, fertig gestellten → Produkte getroffen.

Die Reihenfolge, in welcher die Entscheidungspunkte im Rahmen eines → Projekts durchlaufen werden müssen, wird in der → Projektdurchführungsstrategie festgelegt.

Konventionsabbildung

Konventionsabbildungen stellen den Bezug des → V-Modells zu aktuellen (Quasi-)Standards, Normen und Vorschriften dar.

Eine Konventionsabbildung setzt dazu die Begriffe, die in der Konvention definiert sind, in Beziehung zu dem Begriffssystem des V-Modells.

Produkt

Ein Produkt (im Sinne eines Produkttyps) beschreibt konkrete Produktexemplare, die während eines Entwicklungsprozesses entstehen können.

Produktabhängigkeit

Eine Produktabhängigkeit beschreibt eine Konsistenzbedingung zwischen zwei oder mehreren → Produkten. Dabei kann eine Produktabhängigkeit sowohl innerhalb eines → Vorgehensbausteins als auch zwischen Produkten verschiedener Vorgehensbausteine bestehen.

Man unterscheidet verschiedene Arten von Produktabhängigkeiten (Tailoring-Produktabhängigkeiten, erzeugende Produktabhängigkeiten, relevante Produktabhängigkeiten, strukturelle Produktabhängigkeiten und inhaltliche Produktabhängigkeiten).

Eine Produktabhängigkeit wird als relevant bezeichnet, wenn die betroffenen Produkte sich im Zustand fertig gestellt befinden. Außerdem werden hierbei nur Produktabhängigkeiten betrachtet, die in den ausgewählten Vorgehensbausteinen enthalten sind.

Produktgruppen

Die → Produkte des → V-Modells sind hierarchisch strukturiert. Auf der obersten Ebene befinden sich Produktgruppen. Eine Produktgruppe ist eine Gruppierung einer Menge von inhaltlich eng zusammenhängenden Produkten.

Beispiele für Produktgruppen sind „Angebots- und Vertragswesen“ und „Systementwurf“. Jede Produktgruppe ist eindeutig einem Vorgehensbaustein zugeordnet.

Projekt

Unter einem Projekt versteht man eine einmalige Gesamtheit von koordinierten Aktivitäten mit bestimmten anfangs- und Endpunkten, die von einer Person oder

Organisation mit dem Ziel durchgeführt werden, bestimmte Termin-, Kosten- und Leistungsziele zu erreichen.

Projektdurchführungsstrategie

Eine Projektdurchführungsstrategie legt eine Reihenfolge fest, in der die für das → Projekt relevanten → Entscheidungspunkte durchlaufen werden müssen.

Projektfortschrittsstufe

Eine Projektfortschrittsstufe kennzeichnet einen Zeitpunkt im → Projekt, an dem eine gewisse Entscheidung getroffen wird und somit ein Projektabschnitt beendet wird. Eine Projektfortschrittsstufe wird daher immer erreicht, wenn ein → Entscheidungspunkt erfolgreich durchlaufen wird.

Projektmerkmal

Ein → Projekt wird durch mehrere Projektmerkmale charakterisiert. Jedes Projektmerkmal wird zur Erstellung eines → Anwenderprofils mit einem Wert belegt, der aus einer Menge von möglichen Wertbelegungen ausgewählt werden muss. Beispiele für Projektmerkmale sind „Systemlebenszyklus-Ausschnitt“ oder „Hohe technologische Risiken“.

Projekttypen

Im → V-Modell wird im Wesentlichen zwischen drei unterschiedlichen Projekttypen unterschieden:

- Vergabe und Durchführung von Systementwicklungsprojekten,
- Systemerstellung, Wartung, Pflege und Weiterentwicklung,
- Einführung und Pflege eines organisationsspezifischen Vorgehensmodells.

Für jeden dieser Projekttypen bietet das V-Modell mindestens eine anwendbare → Projektdurchführungsstrategie an.

Rolle

Eine Rolle ist die Beschreibung einer Menge von Aufgaben und Verantwortlichkeiten im Rahmen eines → Projekts und einer Organisation.

Durch die Festlegung von Rollen wird die Unabhängigkeit des → V-Modells von organisatorischen und projektspezifischen Rahmenbedingungen erreicht. Die Zuordnung von Organisationseinheiten und Personen zu den Rollen erfolgt zu Beginn eines Projekts.

Schnittstellenprodukt

Als Schnittstellenprodukt bezeichnet man ein → Produkt, welches zwischen den V-Modell-Projekten von → Auftraggeber und → Auftragnehmer ausgetauscht wird. Die Schnittstellenprodukte sind in der → Auftraggeber-/Auftragnehmer-Schnittstelle festgelegt.

Systemsicherheit

Die Systemsicherheit umfasst die Begriffe Safety und Security.

Safety steht hierbei für die Verfahrens- oder Betriebssicherheit sowie die Aspekte Zuverlässigkeit, Fehlertoleranz und Korrektheit.

Security beschreibt hingegen den Zustand, der die Verfügbarkeit, die Integrität, die Verbindlichkeit und die Vertraulichkeit von Informationen beim Einsatz von IT gewährleistet. Dieser Zustand ergibt sich aus Maßnahmen in der Informationstechnik sowie aus Maßnahmen personeller, materieller und organisatorischer Art.

Dabei ist

- Verfügbarkeit der Zustand, der die erforderliche Nutzbarkeit von Informationen sowie IT-Systemen und Komponenten sicherstellt;
- Integrität der Zustand, der unbefugte und unzulässige Veränderungen von Informationen und an IT-Systemen oder Komponenten ausschließt;
- Verbindlichkeit der Zustand, in dem geforderte oder zugesicherte Eigenschaften oder Merkmale von Informationen und Übertragungstrecken sowohl für die Nutzer verbindlich feststellbar als auch Dritten gegenüber beweisbar sind;

- Vertraulichkeit der Zustand, der unbefugte Informationsgewinnung/-beschaffung ausschließt.

Tailoring

Über die wörtliche Bedeutung des englischen Begriffs hinaus bedeutet Tailoring im Kontext des → V-Modells nicht nur das „Wegschneiden“ von Teilen, sondern auch das „Anpassen“ des → V-Modells. Die Anpassung des V-Modells an ein konkretes → Projekt erfolgt im Normalfall über Hinzunehmen von → Vorgehensbausteinen. Anpassungen innerhalb von Vorgehensbausteinen sind als Ausnahmefall anzusehen.

Je nach Projektfortschritt wird zwischen

- statischem Tailoring, das heißt Tailoring während der Projektinitialisierung und
- dynamischem Tailoring, das heißt Tailoring im weiteren Projektverlauf unterschieden.

Teilaktivität

Eine Teilaktivität gehört zu genau einem → Vorgehensbaustein und ist stets einer → Aktivität zugeordnet. Teilaktivitäten bearbeiten → Produkte und → Themen. Eine Teilaktivität ist eine Beschreibung, wie eine Aufgabe, die typischerweise in einem → Projekt beziehungsweise in einer Organisation anfällt, durchzuführen ist. Teilaktivitäten sind also vergleichbar mit einer Arbeitsanleitung, die geschlossen auszuführen ist, um einen oder mehrere Produktbausteine zu bearbeiten.

Thema

Ein Thema ist eindeutig einem → Produkt zugeordnet, das seinerseits aus beliebig vielen Themen bestehen kann. Ein Thema ist inhaltlicher Natur und in sich abgeschlossen. Die Themen eines Produkts sind als eine Aufzählung der wesentlichen Inhalte des Produkts zu verstehen. Themen werden durch Teilaktivitäten bearbeitet.

V-Modell

Das V-Modell ist ein Leitfaden zum Planen und Durchführen von Entwicklungsprojekten unter Berücksichtigung des gesamten Systemlebenszyklus. Dabei definiert das V-Modell die in einem → Projekt zu erstellenden Ergebnisse und beschreibt die konkreten Vorgehensweisen, mittels derer diese Ergebnisse erarbeitet werden. Darüber hinaus legt das V-Modell die Verantwortlichkeiten der einzelnen Projektbeteiligten fest.

V-Modell-Kern

Der V-Modell-Kern bildet die Basis jedes → Anwendungsprofils. Er legt eine Menge von → Vorgehensbausteinen fest, die in jedem V-Modell-konformen Projekt verwendet werden müssen.

V-Modell-Projekt

Unter einem V-Modell-Projekt versteht man ein → Projekt, das V-Modell-konform durchgeführt wird. Hierbei wird ein Projekt dann als V-Modell-konform bezeichnet, wenn es mindestens die → Vorgehensbausteine und → Produkte des → V-Modell-Kerns beinhaltet sowie jede relevante → Produktabhängigkeit im Rahmen der Entwicklung berücksichtigt.

V-Modell-Referenz

Eine V-Modell-Referenz definiert eine bestimmte Gruppierung der Inhalte des → V-Modells. Die Beschreibungen und Beziehungen der einzelnen → Produkte, → Aktivitäten, → Rollen usw. ändern sich nicht. Sie werden jedoch im Rahmen ihrer Abhängigkeiten neugruppiert und bei Bedarf verkürzt dargestellt. Für verschiedene Anwendungszwecke und Anwender können so angepasste Darstellungen der gleichen Inhalte bereitgestellt werden.

V-Modell-Referenzen werden in der Druckversion des V-Modells in den unterschiedlichen Teilen des V-Modells umgesetzt.

Vorgehensbaustein

Die modulare Einheit des → V-Modells. Das V-Modell ist aus Vorgehensbausteinen zusammengesetzt. Auch wird mithilfe von Vorgehensbausteinen ein projektspezifisches oder → organisationspezifisches Vorgehensmodell erstellt.

Ein Vorgehensbaustein fasst unterschiedliche Aktivitätsbausteine zu einer modularen Einheit zusammen. Indirekt sind ihm somit auch → Produkte zugeordnet, da diese wiederum eindeutig fortlaufenden → Aktivitäten beziehungsweise fertig stellenden Aktivitäten zugeordnet sind.

Vorgehensmodell, organisationspezifisches

Das organisationspezifische Vorgehensmodell dient dazu, ein Verfahren zur Prozessverbesserung in einer Organisation einzuführen, zu etablieren und kontinuierlich zu verbessern. Das hier definierte Vorgehen wird in zwei Einsatzfällen angewandt:

1. Bei der erstmaligen Einführung organisationsweiter Prozessbeschreibungen und deren Umsetzung.
2. Bei der wiederholten Durchführung eines organisationsweiten Prozessverbesserungsprogramms.

Grundlage für den kontinuierlichen Verbesserungsprozess ist das → V-Modell mit all seinen Teilprozessen, → Produkten und → Aktivitäten. Im Rahmen der Einführung eines organisationspezifischen Vorgehensmodells kann das V-Modell an die Organisation angepasst und auch durch organisationseigene Prozesse ergänzt werden. Welche Einheiten dabei zur Organisation gehören, muss am Anfang des Verbesserungsprojekts festgelegt werden.

Ulrich von Hagen
Telefon: 0211 9449-6706
E-Mail: ulrich.von-hagen@lds.nrw.de

Agil und extrem (2)

Praktische Erfahrungen mit agilen Methoden

In der vorangegangenen Ausgabe der LDVZ-Nachrichten wurden in dem Beitrag „Agil und extrem – neue Wege in der Softwareentwicklung“ von Ulrich von Hagen [Hag04] die Prinzipien, Methoden und Praktiken der agilen Softwareentwicklung dargestellt. In dem hier vorliegenden Artikel werden nun einige praktische Erfahrungen mit der Anwendung agiler Methoden in Projekten der LDVZ beschrieben.

Einführung

Seit einigen Jahren nimmt die Geschwindigkeit der Veränderungen im technischen Umfeld, aber auch insbesondere bei den Prozessen der öffentlichen Verwaltung immer mehr zu.

Gerade große und lang laufende Projekte im Bereich der öffentlichen Verwaltung müssen daher noch während des Projektverlaufs auf veränderte fachliche Anforderungen und technische Randbedingungen reagieren können. Schließlich soll das zu erstellende Produkt jene Anforderungen des Kunden erfüllen, die er zum Auslieferungszeitpunkt – also zum Projektende – hat. Da sich alles weiterentwickelt, sind diese Anforderungen nicht immer deckungsgleich mit jenen, die der Kunde zum Projektbeginn definiert.

Nur wenn das Projekt vom Auftraggeber als erfolgreich bewertet wird, ist es auch für den Auftragnehmer erfolgreich.
Jens Coldewey [Col04]

Die Anforderungen, die der Kunde zum Projektbeginn formuliert, stellen häufig lediglich eine mehr oder weniger belastbare Basis für eine anfängliche Projektplanung dar. Als Beispiel sei hier die Zahlung des Urlaubsgeldes an die Landesbeamtinnen und -beamten erwähnt, die zum Beginn eines Projekts der LDVZ zwingend erforderlich war, inzwischen aber abgeschafft wurde. Dies gilt ebenso für den Druck der Lohnsteuerkartenaufkleber, die nicht mehr zu erstellen sind, da die Steuerdaten der Landesbediensteten inzwischen elektronisch an das Finanzamt zu melden sind.

In der Softwareentwicklung werden zurzeit agile Prozesse, wie z. B. eXtreme Programming, SCRUM oder die Crystal-Prozessfamilie (siehe [Hag04]) als besonders geeignet angesehen, wenn Projekte in sog. „instabilen“, d. h. sich schnell ändernden Umfeldern, realisiert werden sollen.

Macht die Dinge so einfach wie möglich, aber nicht einfacher als sie sind!

Albert Einstein

Grundprinzipien

Unmittelbares Feedback

Eines der Grundprinzipien aller agilen Prozesse ist ein unmittelbares Feedback.

Legt man im Projekt die Annahme zugrunde, dass Änderungen an den Anforderungen mit an Sicherheit grenzender Wahrscheinlichkeit erforderlich werden, so ist ein schnelles Feedback, insbesondere von den Realisierern zum Kunden, von essentieller Bedeutung. Gemäß dem Motto „Der Appetit kommt beim Essen“ wird dabei davon ausgegangen, dass die Anforderungen an das Produkt erst im Verlauf der Entwicklung präzisiert werden können, so dass ständige Rückkopplungen des aktuellen Entwicklungsstandes an den Kunden helfen, Fehlentwicklungen bis hin zu einem Produkt, welches für den Auftraggeber unbrauchbar ist, zu vermeiden.

Diese Erfahrung wurde in einigen Projekten gemacht. Zum Projektbeginn wurden Anforderungen an Softwarekomponenten formuliert, deren Realisierung anschließend nicht den tatsächlichen Anforderungen des Kunden entsprach. Zum einen war das auf Missinterpretationen und Missverständnisse zurückzuführen, zum anderen aber auch darauf, dass sich zwischenzeitlich die Anforderungen verändert, d. h. weiterentwickelt hatten.

Inzwischen wird daher angestrebt, die Anforderungen zunächst sehr intensiv zwischen den Fachleuten und der Realisierermannschaft zu diskutieren. Die anschließenden Realisierungen werden eng mit dem Kunden rückgekoppelt, um aufwändige Fehlentwicklungen zu vermeiden.

Einfachheit anstreben

Wer kennt es nicht: Als Softwareentwickler tappen wir immer wieder in die Falle, dass wir uns „tolle Dinge“ ausdenken, die weit abgehoben von den tatsächlichen Notwendigkeiten sind. Bevor man also daran geht, die Sonderfälle zu lösen, sollte man zunächst einmal die zwingend notwendige Grundfunktionalität realisiert haben. Es ist daher für ein Entwicklungsteam und auch für den Kunden wichtig, dass

man sich immer wieder die Frage stellt: Ist dies oder jenes wirklich erforderlich? So würde es z. B. bei einem System zur Verwaltung von Schüler(inne)n wenig nutzen, wenn man zwar einerseits den mehrfachen Wechsel einer Schülerin bzw. eines Schülers von einer Klasse in eine andere Klasse tiefgehend analysiert und auch bereits umgesetzt hat, aber andererseits noch nicht in der Lage ist, die Stammdaten zu einer Schülerin bzw. einem Schüler überhaupt zu erfassen. Es geht also darum, nur die Anforderungen zu realisieren, die der Kunde wirklich hat.

Inkrementelle Veränderung

Bei der Idee der inkrementellen Entwicklung geht man davon aus, dass es besser im Sinne eines geringeren Aufwandes und eines geringeren Projektrisikos ist, das geplante Produkt nicht in einem einzigen Wurf zu realisieren, sondern mit einem kleineren beherrschbaren „Kernsystem“ zu beginnen, dem anschließend so lange Eigenschaften hinzugefügt werden, bis die Anforderungen erfüllt sind.

Zu jedem Zeitpunkt lässt sich dabei begutachten, welche Komponenten bereits funktionieren und getestet sind.

In einigen kleineren Projekten konnte dieser Ansatz erfolgreich eingesetzt werden, um nach und nach den Funktionsumfang des Produkts zu erweitern und das Feedback des Kunden unmittelbar einzuholen. In anderen Projekten, in denen eine umfangreiche Kernfunktionalität vorhanden sein muss, um sinnvoll den produktiven Einsatz zu beginnen (ein Flugzeug ohne Landefunktion ist nicht sinnvoll einsetzbar), kann ebenfalls inkrementell bzw. iterativ entwickelt werden, nur gelangen viele dieser Inkremente nicht in die Produktion, so dass hier das Feedback des Kunden aus dem produktiven Einsatz fehlt. Hier sollte der Kunde noch enger in die Testaktivitäten einbezogen werden.

*Nie aber weiß man genau,
was man denkt,
bevor man nicht versucht hat,
darüber zu sprechen.*

Kurt Marti

Veränderungen wollen

Änderungen an fachlichen Anforderungen oder technischen Rahmenbedingungen oder sonstigen wichtigen Einflussgrößen stellen zunächst immer ein Risiko für die Projektdurchführung dar. Häufig haben Änderungen (meist negative) Auswirkungen auf Termin und Budget.

Auf der anderen Seite formuliert der Kunde seine Änderungsforderungen für gewöhnlich nicht, um den realisierenden Auftragnehmer damit zu ärgern oder sonst wie in Bedrängnis zu bringen. Vielmehr verspricht er sich hierdurch einen Mehrwert, einen Wettbewerbsvorteil oder auch einfach nur die Sicherstellung seiner eigenen Wettbewerbsfähigkeit.

Somit müssen Änderungen auch als Chance gesehen werden, den Erfolg des Projekts zu sichern. Das Vermeiden von Änderungen, mit dem Ziel, das Projekt möglichst exakt nach einem zu Beginn vorgegebenen Plan durchzuführen, führt fast unweigerlich dazu, ein für den Kunden nicht zufrieden stellendes Produkt zu entwickeln.

In der DIN 69901 wird in der Definition des Begriffes Projekt auch die „Einmaligkeit der Bedingungen in ihrer Gesamtheit“ gefordert. Diese „Einmaligkeit der Bedingungen“ führt immer zu dem Risiko, dass sich das Projekt ganz anders verhält als geplant. Streng genommen kann ein Vorhaben ohne Risiko, in dem alle Parameter exakt kontrolliert werden können, kein „Projekt“ im Sinne der DIN 69901 sein.

Also muss es das Ziel einer effektiven Projektsteuerung sein, Methoden zu finden, die es erlauben, Änderungen während des Projektverlaufs zu integrieren.

Hier muss sich letztendlich ein vernünftiger Umgang mit Änderungen einspielen. Es kann nämlich weder darum gehen, dass sämtliche Änderungswünsche des Kunden – mit u. U. fadenscheinigen Argumenten – abgeblockt werden, noch darum, dass jeglicher bekannt werdende Änderungswunsch „ohne Rücksicht auf Verlust“ sofort umgesetzt wird.

Der Kunde sollte sich klar machen, dass aufgrund seines begrenzten Budgets ggf. nicht alle Wünsche umgesetzt werden können und dass er daher insbesondere im Hinblick auf eine inkrementelle Entwicklung des Systems eine Priorisierung seiner Wünsche und Anforderungen vornehmen muss. Diese Priorisierung kann sich selbstverständlich im Laufe der Zeit ändern. Wenn jedoch bestimmte Wünsche eine höhere Priorität erhalten, so sollte auch klar sein, was dafür in der Prioritätenliste weiter nach hinten wandert.

Das Entwicklungsteam hat zwar die Aufgabe aufzuzeigen, wo u. U. Risiken bei der Umsetzung der einzelnen Kundenwünsche zu vermuten sind und mit welchen Realisierungsaufwänden jeweils zu rechnen ist, aber letztendlich entscheidet der Kunde, was realisiert werden soll – schließlich bezahlt er ja auch dafür. Wenn sich der Kunde z. B. im Projektverlauf entschließt, dass die Ausgabe bestimmter Daten nun nicht mehr wie ursprünglich festgelegt nach MS Word, sondern nach MS EXCEL erfolgen soll, so sollte das Entwicklungsteam diese Änderung „ohne Murren“ entgegen nehmen und genauso gewissenhaft umsetzen, wie es bei Anforderungen der Fall ist, die seit Projektbeginn bekannt sind. Soweit hiermit Risiken und Mehraufwände verbunden sind, gehört es selbstverständlich dazu, dass diese dem Kunden erläutert werden.

Im Projekt Obelix (siehe [And04]) wird versucht, möglichst rasch auf Änderungen der Anforderungen reagieren zu können, indem in den wöchentlichen

chen Projektbesprechungen die Prioritäten der zu realisierenden Funktionen überprüft und ggf. angepasst werden. Damit ist es möglich, ein Produkt zu entwickeln, welches zum Projektende die Anforderungen des Kunden trifft und ihm eine maßgeschneiderte Lösung für seine Geschäftsprozesse liefert.

Eine vertrauensvolle Zusammenarbeit zwischen Kunden und Auftragnehmer ist hierbei unumgänglich (siehe auch Abschnitt „Play to win“ anstatt „Play not to loose“).

Qualitätsarbeit

„Bananensoftware reift beim Kunden“ heißt es in des Softwareentwicklers Volksmund. Andererseits streben wir an, „dass der Kunde zurückkommt und nicht das Produkt“.

Was Qualität ist, bestimmt dabei der Kunde. Wichtig ist hierbei, dass das Produkt für den täglichen Gebrauch tauglich ist und sich die darin abgebildeten Geschäftsprozesse auch tatsächlich durchführen lassen und dass es dem Kunden einen echten Mehrwert bietet.

Die Qualität der Produkte wird in der LDVZ groß geschrieben und durch eine Reihe konstruktiver sowie analytischer Maßnahmen sichergestellt. Standards und Richtlinien auf der einen Seite und mehrstufige Testkonzepte auf der anderen Seite sorgen dafür, dass die hohen Qualitätsanforderungen der Kunden erfüllt und nachgewiesen werden können.

**Aufrichtigkeit
ist wahrscheinlich die
verwegenste Form der Tapferkeit.**
William Somerset Maugham

„Play to win“ anstatt „Play not to loose“

„We value close customer collaboration over contract negotiation“ heißt es im Agilen Manifest.

Grundannahme ist hier also das gemeinsame Spielen auf Sieg, und nicht das häufig anzutreffende Schwarze-Peter-Spiel.

Jens Coldewey geht in seinem Beitrag in der Zeitschrift Objektspektrum [Col04] sogar soweit, dass Projekte, die in einer Atmosphäre des Misstrauens durchgeführt werden, zwangsläufig unter ihrem eigentlichen Potenzial zurückbleiben müssen. Der Auftragnehmer wird versuchen, keine Fehler zu machen, und den Projektauftrag und die Anforderungen buchstabengetreu umzusetzen. Hierdurch wird der Auftraggeber fast zwangsläufig ein Produkt erhalten, was seine Anforderungen nicht vollständig erfüllt. Hier wird es sehr schwer, Missverständnisse zu vermeiden, direkte und insbesondere offene Kommunikation ist in einem solchen Projektumfeld nicht mehr möglich.

Und die oben beschriebenen Probleme sind die Hauptursachen für gescheiterte Projekte, da Projekte bekanntlich selten an technologischen Problemen scheitern, sondern in den meisten Fällen an Problemen im Bereich der Kommunikation und des Projektklimas.

Hinsichtlich der vertraglichen Auftragsabwicklung bei der Pflege von Softwaresystemen hat sich inzwischen – sowohl bei Aufträgen im Rahmen von vereinbarten Mittelkontingenten als auch bei darüber hinausgehenden Aufträgen – ein Vorgehen bewährt, bei dem, ausgehend von den vorgetragenen Kundenwünschen, diese zunächst hinreichend abgeklärt werden, so dass eine Aufwandsabschätzung zu einzelnen Erweiterungs-/Anpassungswünschen erfolgen kann. D. h. in entsprechenden Angeboten werden einzelne Positionen mit ihren jeweiligen Kosten aufgeführt. Diese Aufwandsabschätzung dient dann dem Kunden dazu, zu entscheiden, welche seiner Wünsche er umsetzen lassen will. Für den Kunden eröffnet sich damit in

Kenntnis des jeweiligen Realisierungsaufwands die Möglichkeit, bestimmte Erweiterungen und Anpassungen (zunächst) nicht zu beauftragen. Ein solches Vorgehen bringt Vorteile für beide Seiten: Der Kunde erhält diejenigen Erweiterungen und Anpassungen, die für ihn den größten Mehrwert bringen; das Entwicklerteam arbeitet ergebnisorientiert an Software, die vom Kunden tatsächlich benötigt wird.

**Nicht Sprüche sind es,
woran es fehlt, die Bücher sind
voll davon.
Woran es fehlt, sind Menschen,
die sie anwenden.**
Epiktet

Praktiken

Kunde vor Ort

In einem Projekt war zum Projektstart das Team auf zwei Standorte verteilt, ein Teil arbeitete beim Kunden, ein anderer im LDS NRW. Auftragsübergaben und fachliche Abstimmungen fanden bei Bedarf in Besprechungen, telefonisch oder per E-Mail statt.

Bei der täglichen Arbeit zeigte sich aber immer häufiger, dass der Gesprächsbedarf wesentlich höher war als angenommen. Die Entwicklung einer gemeinsamen Sprache und Begriffswelt, in der sich die Vertreter der Kundenseite mit den Realisierern unterhalten können, hat sich auf der einen Seite als sehr aufwändig, andererseits aber auch als essentiell für das Projekt herausgestellt.

Im Projektverlauf ist es dann gelungen, auch alle Mitglieder des Kunden-Projektteams in dem Gebäude unterzubringen, in dem bislang nur die Realisierermannschaft des LDS NRW untergebracht war.

Diese Maßnahme hat sich sehr positiv auf das gesamte Projekt ausgewirkt. Liegen zwischen den Gebäuden auch nur wenige Kilometer, so beeinflusst diese Distanz doch ganz erheblich die direkte Kommunikation zwischen den Projektbeteiligten.

Inzwischen finden alle Abstimmungen zwischen Realisierern und dem Kunden im Rahmen der direkten Kommunikation statt. Missverständnisse werden hierdurch drastisch reduziert.

Auch wenn kein Kundenvertreter unmittelbar im Projekt mitwirkt, so sollte doch nach Möglichkeiten gesucht werden, die Kundenseite stärker einzubeziehen und ein möglichst zeitnahes Feedback zu gewährleisten.

Auf jeden Fall sollte auf der Kunden- seite ein kompetenter Ansprechpartner benannt werden, der in der Lage ist, die Kundenwünsche aufzugreifen und innerhalb der Kundenorganisation(en) abzustimmen. Dieser Ansprechpartner hat dann auch die Aufgabe, bei unklaren oder sich widersprechenden Anforderungen kurzfristig eine Klärung herbeizuführen.

Bei der Pflege von Softwaresystemen¹⁾ hat es sich zudem vielfach bewährt, zwischen der Abstimmung der verbal formulierten Anforderungen und der konkreten Umsetzung eine zusätzliche Abklärung der Kundenwünsche auf der Grundlage von Maskenentwürfen vorzunehmen. Es zeigt sich bekanntlich ja immer wieder, dass auch die noch so gewissenhaft formulierten schriftlichen Anforderungen nicht so klar sind, dass sie ohne weitere Rückkopplung mit dem Kunden umgesetzt werden können.

1) Zu nennen wäre hier z. B. die Weiterentwicklung des Verfahrens WAVE, wo sich der Kunde, die Vertretung des Landes Nordrhein-Westfalen beim Bund, an einem anderen Standort – nämlich Berlin – befindet. (Zum Verfahren WAVE vgl. auch Artikel „WAVE – Eine umfassende Unterstützung für die Veranstaltungsplanung und -durchführung“ in dieser Ausgabe der LDVZ-Nachrichten.)

***Prognosen sind schwierig,
vor allem wenn sie die Zukunft
betreffen***

Mark Twain

Planungsspiel

In einem Projekt wurde ein Planungs- Ansatz gewählt, nach dem der Kunde zunächst seine Anforderungen in gro- ben Umrissen – also noch nicht vollständig – formuliert. Anschließend wird der Realisierungsaufwand durch die Realisierer im Rahmen einer Schätzklausur geschätzt. Schlussend- lich entscheidet der Kunde dann durch die Priorisierung seiner Anforderun- gen, was in welcher Reihenfolge reali- siert wird, und dadurch auch, in wel- cher Reihenfolge er seine Anforderun- gen detailliert beschreiben und an die Realisierung übergeben muss.

Hierdurch wird sichergestellt, dass ers- tens die Komponente, an der aktuell gearbeitet wird, wichtiger ist als alle noch zu realisierenden Komponenten und dass zweitens alle bereits realisiert- en Komponenten wichtiger sind als alle, die noch offen sind. Hierdurch wird gewährleistet, dass zu jedem Zeit- punkt die aktuell wichtigsten Aktivitä- ten durchgeführt werden und dass zu jedem Zeitpunkt die wichtigsten Funk- tionen in das System eingebaut sind.

Fairerweise muss hier erwähnt werden, dass es für die meisten Anwendungs- fälle eine definierte Mindestfunktio- nalität gibt, ohne die ein sinnvoller Ein- satz des Systems nicht möglich ist. Ein Personalabrechnungsverfahren, wel- ches keine Zahlungen auslösen kann, ist für den Kunden nicht nutzbar.

Anders sieht dies natürlich bei Erwei- terungen und Anpassungen eines Sys- tems aus, das bereits im Einsatz ist. Hier kann bereits durch das Hinzufü- gen einzelner wichtiger Funktionen ein deutlicher Mehrwert für den Kunden erzielt werden. So konnte im Umfeld

des Seminarverwaltungssystems für die Fortbildungsakademie Herne die Präsentation des Seminarangebots da- durch deutlich verbessert werden, dass nunmehr verschiedene Suchmöglich- keiten unterstützt werden (vgl. hierzu auch Artikel „Seminarverwaltungssys- tem für die Fortbildungsakademie Her- ne“ in den LDVZ-Nachrichten 2/2004).

Kurze Releasezyklen

In einem größeren Projekt der LDVZ mit einem Projektteam von 40 Personen wird pro Arbeitswoche ein Aufwand von einem ganzen Personenjahr mit den sich daraus ergebenden Kosten geleis- tet. Es ist daher sehr wichtig, durch kur- ze Entwicklungszyklen Fehlentwicklun- gen frühzeitig zu erkennen. In dem Buch „Agile Softwareentwicklung im Großen“ [Eck04] vertritt die Autorin Jutta Eckstein hierzu sogar die These, dass die Releasezyklen umso kürzer sein sollten, je größer das Projekt ist, da Fehlentwicklungen sehr schnell sehr hohe Aufwände und Kosten erzeugen.

Im oben genannten Projekt wird aus diesem Grunde wöchentlich ein aktuel- les Build der gesamten Software herge- stellt, um die Qualität und den Fort- schritt, d. h. den Funktionsumfang di- rekt an der Software messen zu können.

Hierbei handelt es sich zunächst um in- terne Integration-Builds, die noch nicht für einen produktiven Einsatz geeignet sind, weil sie noch nicht die erforderliche Minimalfunktionalität für einen Produktionsstart enthalten.

Wie bereits weiter oben erwähnt, ist es bei der Pflege von Softwaresystemen vielfach eher möglich (und auch vom Kunden gewünscht), dass bestimmte Erweiterungen und Anpassungen des Systems in kürzeren Zyklen in die Pro- duktion überführt werden.

Akzeptanztests

Akzeptanztests spielen in agilen Pro- jekten eine wichtige Rolle. In einem Kundenprojekt werden dabei auf der Basis der fachlichen Anforderungen

(funktionale Abnahmekriterien) fachliche Testfälle beschrieben, denen das System genügen muss.

Die fachlichen Testfälle werden dabei vom Kunden spezifiziert, da hier das gesamte Fachwissen verfügbar ist. Diese fachlichen Testfälle werden dann technisch in Testdatenkombinationen sowie Testabläufe übersetzt, mit denen anschließend die Software getestet werden kann. Bei „neuen“ Komponenten werden diese Tests manuell durchgeführt.

Die oben beschriebenen wöchentlichen Versionen der Software werden anschließend mit Hilfe des Test-Roboters von Rational IBM [Kür02] mit allen bis zu diesem Zeitpunkt automatisierten Testfällen getestet, um sicherzustellen, dass durch unbeabsichtigte Seiteneffekte keine neuen Fehler unbemerkt in bereits fertig gestellten – d. h. erfolgreich getesteten – Komponenten entstehen.

Testgetriebene Entwicklung

Die Idee der testgetriebenen Entwicklung ist (wie bereits in [Hag04] beschrieben), die zu erstellende Software bereits während der Entwicklung ständig zu testen und sogar die dafür erforderlichen Testtreiber bereits vor der eigentlichen Programmierung zu erstellen.

Dadurch wird zum einen ständig die Lauffähigkeit der Software überprüft, und zum anderen durch die Formulierung der Kundenanforderungen in Form eines Testtreibers sichergestellt, dass tatsächlich genau diese Anforderungen realisiert werden.

In einem Projekt in der LDVZ werden dabei laufend während der Entwicklung automatisierte Unit-Tests durchgeführt. Der Nachweis des erfolgreich durchgeführten Unit-Tests ist dabei zwingend erforderlich, um das entsprechende Modul an die nächste Teststufe abgeben zu können.

Refactoring

Bei der Pflege von Softwaresystemen ist ein Refactoring bzw. eine Restruktu-

rierung z. B. dann angezeigt, wenn Anpassungen und Erweiterungen für einzelne Kunden durchgeführt wurden und man Gefahr läuft, dass sich das Softwaresystem in eine Vielzahl von Varianten aufspaltet. In solchen Fällen hat es sich oftmals bewährt, die entstandenen Varianten wieder zu einem einheitlichen System zusammenzuführen und die spezifischen Aspekte der einzelnen Kunden über eine geeignete Parametrisierung zu berücksichtigen. Für das Verfahren WAVE wurde z. B. aufgrund eines Kundenwunsches neben der ursprünglich im MS Access-Umfeld genutzten Jet Engine auch der MS SQL-Server als Datenbanksystem verwendet, woraus zunächst zwei unterschiedliche WAVE-Varianten resultierten. Durch eine entsprechende Restrukturierung wurden diese beiden Varianten inzwischen zusammengeführt, so dass künftig nur noch ein System zu pflegen ist, das beide Datenbanksysteme unterstützt.

Im Projekt Obelix haben einige erforderliche Änderungen an Anforderungen zu Überarbeitungen des Software-Designs geführt. Dabei wurden von den Realisierern in dedizierten Refactoring-Zyklen diese Design-Änderungen in die Software eingearbeitet, ohne die Funktionalität nach außen zu verändern. Dies wurde durch automatisierte Tests (siehe [Kür02]) nachgewiesen.

Programmieren in Paaren

Im Zusammenhang mit der Entwicklung des Verfahrens EIFEL (Eignungstest für Feuerwehrleute)²⁾ wurden positive Erfahrungen mit dem Programmieren in Paaren gemacht. In diesem Fall arbeiteten ein erfahrener Programmierer des LDS NRW und ein Praktikant sehr effektiv und nutzbringend zusammen. Zudem war der Lerneffekt, der durch diese Art der Programmierung erzielt wurde, beachtlich (vgl. hierzu auch den Artikel „Oft verkannt – der Praktikant“ in dieser Ausgabe der LDVZ-Nachrichten).

²⁾ siehe auch Artikel „EIFEL – Eignungstest für Feuerwehrleute“ in dieser Ausgabe der LDVZ-Nachrichten

Auf den Nutzen, den das Programmieren in Paaren auch für zwei erfahrene Entwickler bringen kann (z. B. Verbesserung der Software-Qualität durch das permanente Code-Review), wurde bereits in [Hag04] hingewiesen.

In einem anderen Projekt wurde es den Programmierern freigestellt einzeln oder in Paaren zu entwickeln. Dabei haben sich keine Programmierpaare gebildet. Zum einen ist für das Pairprogramming die Atmosphäre im Team, die „Chemie“ zwischen den Entwicklern, von entscheidender Bedeutung. Programmierer, die Vorbehalte gegenüber dem anderen haben, werden als Paar sicherlich nicht produktiver arbeiten können als einzeln. Zum anderen bedeutet Pairprogramming einen ziemlich weitgehenden Paradigmenwechsel, da es ein hohes Maß an bisher vielfach nicht gekannter Offenheit erfordert.

Gut haben sich hingegen in einem anderen Projekt Paare aus Entwicklern verschiedener Aufgabenbereiche bewährt, in diesem Fall Paare aus Software-Designern und Programmierern. Dabei wurde von diesen Paaren das vom Designer entwickelte Design direkt im Programm ausprobiert. Dadurch konnten zum einen sehr schnell Missverständnisse hinsichtlich der Interpretation des Designs in der Realisierung geklärt und zum anderen die Umsetzbarkeit von neuen Designobjekten überprüft werden.

Wer sich das Alte noch einmal vor Augen führt, um das Neue zu verstehen, der kann anderen ein Lehrer sein.

Konfizius

Projekt-Retrospektiven

Auch in gut eingespielten Teams macht es – unabhängig von der Teamgröße und der Art der Entwicklungsaufgaben – Sinn, sich von Zeit zu Zeit Gedanken über die eigenen Vorgehensweisen zu machen. Dabei ist es natür-

lich wichtig, dass ausgehend von der Erörterung aufgetretener Probleme und Reibungsverluste konkrete Verbesserungsmaßnahmen festgelegt werden.

In einem größeren Projekt der LDVZ wurden im Projektverlauf bei Bedarf immer wieder Retrospektiven durchgeführt, um die Vorgehensweisen (Prozesse, Risikobetrachtungen, Planungsgrundlagen, etc.) zu überprüfen und die Durchführung des Projekts an die veränderten Bedürfnisse anzupassen. So wurden beispielsweise Workshops durchgeführt, in denen die Effektivität und Effizienz der Übergabewege der zu erstellenden Teilprodukte zwischen verschiedenen Teilprojektteams hinterfragt wurde und diese Prozesse ggf. den Erkenntnissen angepasst wurden.

Der eindeutige Vorteil einer solchen projektbezogenen Reflektion – gegenüber einem Verbesserungsprozess auf Ebene des organisationsspezifischen Vorgehensmodells – liegt darin, dass die Verbesserungen sich an der Projektwirklichkeit orientieren und bereits im nächsten Projektabschnitt bzw. im anschließenden Folgeprojekt Nutzen entfalten können.

Zusammenhang und Abgrenzung zwischen V-Modell und agilen Prozessen³⁾

Um klar zu sehen, genügt oft ein Wechsel der Blickrichtung.

Antoine de Saint-Exupéry

In der LDVZ wird Software nach dem ISO 9000:2000 zertifizierten hauseigenen Vorgehensmodell ABS (Anwendungsbereitstellungsstandard) entwickelt. Dieses Vorgehensmodell ist sehr stark an das V-Modell 97 angelehnt.

3) Wir beziehen uns hierbei auf das V-Modell 97. Das vor kurzem erschienene neue V-Modell XT greift agile Vorgehensweise ausdrücklich auf (vgl. hierzu den Artikel „Das neue V-Modell XT“ in dieser Ausgabe der LDVZ-Nachrichten).

In diesem Abschnitt wollen wir anhand der Aktivitätenbereiche des ABS aufzeigen, in welcher Weise agile Praktiken ergänzend hilfreich sein können, d. h. wir betrachten hier nun einige Erfahrungen mit agilen Vorgehensweisen vor dem Hintergrund eines klassischen Vorgehensmodells.

Auftragsanalyse

Startpunkt des ABS ist die Auftragsanalyse, in der die Projektziele formuliert, die Randbedingungen beschrieben und eine erste sehr grobe Planung von Aufwand, Ressourcen und Endtermin vorgenommen werden.

Angesichts sich schnell verändernder Anforderungen muss in der Auftragsanalyse der Projektrahmen abgesteckt werden. Da es in den meisten Fällen aber nicht möglich sein wird, zu Projektbeginn zu einem vollständigen und unverrückbaren Katalog der Kundenwünsche zu gelangen, sollte die Detaillierung, Fortschreibung und Priorisierung der Kundenwünsche während des Projektverlaufs im Rahmen des Planungsspiels (s. o.) erfolgen.

Projektmanagement

Nach der Auftragsanalyse ist das Submodell PM (Projektmanagement) der Einstieg in das Projekt. Im Projektstrukturplan (PSP) werden alle Aktivitäten beschrieben, die im Projekt durchzuführen sind. Darauf basieren dann Ressourcen- und Ablaufplanung und daraus resultierend die Terminplanung. Auf die Aktivitäten des PSP verweisen das Konfigurations- und Änderungsmanagement sowie der QS-Plan sowie die integrierte Projektsteuerung mit der Messung des Fortschrittsgrades.

In der Philosophie agiler Entwicklung scheinen sich diese Aspekte zu widersprechen, da man ja gerade annimmt, dass im Projektverlauf Änderungen erforderlich werden, und dadurch eine detaillierte Planung der durchzuführenden Aktivitäten vorab unmöglich ist.

Andererseits wird es ohne eine Übersicht der durchzuführenden Aktivitäten (PSP) unmöglich sein, den Gesamtaufwand, den Ressourcenbedarf und die Auslieferungstermine zu prognostizieren. Dies hätte zur Folge, dass sich nichts mehr über die Wirtschaftlichkeit des Projekts, den ROI (Return of Investment) und den Vergleich mit Alternativen aussagen lässt.

Unter diesen Voraussetzungen wird man aber nur noch schwer einen Auftraggeber finden, der für ein solches Vorhaben Zeit, Geld und Personal zur Verfügung stellt.

Wie lässt sich nun dieser scheinbare Widerspruch auflösen?

Zum einen ist es erforderlich, zu Beginn eine initiale Planung des Projekts wie oben beschrieben durchzuführen, um Durchführungsentscheidungen zu erhalten, sich von Alternativen abzugrenzen und Budget-, Zeit- und Personalbedarf abzuschätzen.

Dabei werden die Aktivitäten nur so detailliert geplant, wie es die Genauigkeit der vorliegenden Informationen erlaubt: „Planung so viel wie nötig und so wenig wie möglich“, denn Projektmanagement kostet Zeit und Geld. Im Projektmanagement selbst liegt kein Mehrwert für den Kunden, es soll lediglich dazu dienen, die Projektziele „on time und on budget“ zu erreichen (siehe [Ger03]).

Wurde das Projekt gestartet, so werden die nächsten Schritte so detailliert geplant, wie nötig. Das bedeutet, dass die nächste Iteration sehr detailliert geplant und deren Durchführung verfolgt wird, während zukünftige Entwicklungsschritte weniger konkret geplant werden. Hierdurch werden hohe Überarbeitungsaufwände der Planung aufgrund von Änderungen vermieden.

Auf Änderungen kann nun agil auf verschiedene Arten reagiert werden: Soll eine Funktion realisiert werden, die bis-

lang nicht vorgesehen war, so wird bei gleich bleibendem Budget vom Kunden entschieden, welche Funktion im Gegenzug nicht mehr realisiert wird, oder aber dass das Budget und der Terminplan entsprechend aufgestockt werden.

Wichtig ist dabei insbesondere in agilen Projekten, dass trotz oder gerade wegen der durchzuführenden Änderungen die Planungsunterlagen jederzeit auf dem neuesten Stand sind, so dass eine Aussage darüber, wie der erreichte Fortschrittsgrad bezogen auf die aktuellen Planungsgrundlagen ist und ob damit die Projektziele erreicht werden können, möglich ist.

Typischerweise sind Softwaresysteme nicht mit der erstmaligen Überleitung in die Produktion abgeschlossen, sondern werden anschließend über Jahre oder auch Jahrzehnte geändert und erweitert. Hierbei bietet eine produktorientierte Denkweise – als Alternative zur Projektorganisation – eine Reihe von Vorteilen, da bei einer solchen Herangehensweise der (wirtschaftliche) Erfolg des Produkts über seinen gesamten Lebenszyklus betrachtet wird (vgl. hierzu auch [Col05]).

Bei kleineren Pflegeaufgaben zeigte sich auch, dass die Beteiligung des Projektleiters insbesondere im Hinblick auf die Analyse und Präzisierung der Kundenwünsche zurückgenommen werden kann, da das Entwicklerteam aufgrund seiner intimen Kenntnisse des vorhandenen Softwaresystems viel eher in der Lage ist, die von den Kunden geäußerten Erweiterungs- und Anpassungswünsche zu verstehen und einzuordnen.

Konfigurations- und Versionsmanagement

Das Submodell KM ist im Projekt Dreh- und Angelpunkt aller weiteren Aktivitäten.

Hier wird definiert, welche Konfigurationen existieren und aus welchen Komponenten sie bestehen. Aktivitäten im Projektplan beziehen sich also auf definierte Konfigurationen. Beabsichtigte Änderungen verändern bestimmte

Konfigurationen, d. h. sie erzeugen eine neue Version der Konfiguration, in der neue Bestandteile hinzukommen oder wegfallen oder verändert wurden.

Gerade in einem großen Projekt in einem instabilen Umfeld ist ein effektives Konfigurationsmanagement unverzichtbar. Durch die Projektgröße entstehen eine große Zahl an Teilprodukten, die anschließend durch Änderungen in vielen Versionen existieren. In einem Projekt der LDVZ sind dies mehrere 100 000 Teilprodukte. Ohne ein funktionierendes Konfigurationsmanagement ist es in diesem Projekt unmöglich nachzuvollziehen, aufgrund welcher fachlichen Änderung welcher Anforderung das Modul XYZ in der Version V erstellt wurde, und mit welchen anderen Modulen, Datenbanken, Testdaten, etc. in welchen Versionen es zusammenwirken muss, um eine lauffähige Version der Software zu bilden. Da die Entwicklung in dem oben erwähnten Projekt auf vielen verschiedenen Plattformen (verschiedene Repositories, Großrechner, Datenbank, ...) stattfindet, war es bislang nicht möglich, das Konfigurationsmanagement mit einem einzigen Tool abzubilden. Hier wurden für jede Plattform eigene Lösungen entwickelt, die anschließend über das Tool ClearQuest (Rational IBM) zusammengehalten werden.

Gerade in der agilen Softwareentwicklung ist ein effektives Konfigurationsmanagement unerlässlich, da nur dadurch möglich ist, festzustellen, welche Objekte von einer Änderung betroffen sind und welche Versionsstände der Objekte zu einem bestimmten Release gehören.

Änderungsmanagement

Im V-Modell 97 und im ABS gehört das Änderungsmanagement zum Submodell KM. Dies macht durchaus Sinn, weil durch die beabsichtigten Änderungen Konfigurationen geändert und neue Versionen einzelner Komponenten erzeugt werden.⁴⁾

4) Im neuen V-Modell XT wurde das Problem- und Änderungsmanagement als eigenständiger Vorgehensbaustein ausgestaltet.

In einem agilen Umfeld, ist ein effektives Änderungsmanagement möglicherweise das stärkste Steuerungsinstrument überhaupt. Hier muss entschieden werden, welche Änderungen durchgeführt werden sollen, und welche nicht, und in welches Release diese Änderungen integriert werden sollen. Fragen wie, „Müssen Schreibfehler auf Dialogmasken sofort behoben werden, oder sollte sich das Projektteam zunächst auf gravierende Fehler konzentrieren und die 'Kleinigkeiten' in einem späteren Release beheben?“, werden hier gestellt und entschieden.

Ein funktionierendes Änderungsmanagement geht also Hand in Hand mit einem leistungsfähigen Requirements-Management.

Dabei hat sich im oben erwähnten Projekt ein zweimal wöchentlich tagendes Gremium – selbstverständlich unter Beteiligung des Kunden – bewährt, welches über die Durchführung und die Terminierung der Änderungsanträge entscheidet.

Der Prozess für das Änderungsmanagement und die gesamte Verwaltung der Arbeitsaufträge wurde in diesem Projekt mit dem Tool ClearQuest (Rational IBM) abgebildet. Hierbei handelt es sich im Wesentlichen um eine relationale Datenbank mit Workflowkomponenten für die Verwaltung der Fehlermeldungen und Änderungsanträge.

Tatsachen hören nicht auf zu bestehen, weil sie unbeachtet bleiben.

Aldous Huxley

Risikomanagement

Risiken bedrohen den Erfolg eines jeden Projekts. Die Kunst liegt hier darin, mit den Risiken intelligent umzugehen, und sie aktiv in die Projektsteuerung einzubeziehen.

Das Risikomanagement, so wie es z. B. auch in ABS beschrieben ist, beschäf-

tigt sich mit der Identifikation, der Analyse, der Maßnahmenplanung und der Risiko- und Maßnahmenkontrolle.

Gerade in Projekten in einem instabilen Umfeld kommt dem Risikomanagement eine besondere Bedeutung zu, da die Risiken, dass es durch Änderungen zu massiven Überarbeitungsaufwänden kommt, und dadurch das Projekt „out of time and budget“ gerät, inhärent sind.

Hier bedarf es also eines effektiven Risikocontrollings und einer engen Verzahnung mit dem Projektmanagement, um z. B. sicherzustellen, dass kritische Komponenten zuerst gebaut und intensiver getestet werden, während andere, die für den Geschäftserfolg nicht überlebenswichtig sind, erst später erstellt werden.

Dies wird durch die oben beschriebenen Praktiken „Planungsspiel“, „Kurze Releasezyklen“, „Testgetriebene Entwicklung“ und die ständigen „Akzeptanztests“ sehr gut unterstützt.

Qualitätssicherung

Insbesondere die laufenden Qualitätssicherungsmaßnahmen von einem Entwicklungsschritt zum nächsten, die auch schon das V-Modell kennzeichnen, sorgen für eine hohe Qualität und einen geringen Entwicklungsaufwand.

Mögliche Fehler werden dabei durch konstruktive Qualitätssicherungsmaßnahmen, wie z. B. Programmierstandards und ein ingenieurmäßiges systematisches Vorgehen, von vorneherein vermieden. Darüber hinaus werden durch analytische Qualitätssicherungsmaßnahmen, wie z. B. Entwurfs-Reviews und ein mehrstufiges Testkonzept, Fehler möglichst frühzeitig gefunden. Der Aufwand für die Behebung von Fehlern wird umso größer, je später sie gefunden werden.

Im ABS wie auch im V-Modell werden dabei laufend Realisierungsprodukte (z. B. Softwarekomponenten) gegen die entsprechenden Entwurfsprodukte (z. B. Softwarearchitektur) geprüft, um

sicherzustellen, dass das letztendlich ausgelieferte Produkt den dokumentierten Anforderungen entspricht.

Die weiter oben beschriebenen agilen Praktiken „Testgetriebene Entwicklung“ und ständige „Akzeptanztests“ unterstützen dies in besonderem Maße.

Beständig ist nur der Wandel.

Heraklit

Zusammenfassung

Die Softwareentwicklung wird in der LDVZ mit Hilfe des V-Modell-ähnlichen Vorgehensmodells ABS durchgeführt. In einem instabilen Umfeld, welches häufige Änderungen an den Anforderungen mit sich bringt, werden diese „klassischen“ Methoden gut durch agile Methoden ergänzt.

Hierbei hat sich insbesondere die intensive direkte Kommunikation durch den „Kunden vor Ort“ als extrem wichtig für die Projekte herausgestellt. Die kurzen Releasezyklen, die ein häufiges und unmittelbares Feedback an den und von dem Kunden ermöglichen, die testgetriebene Entwicklung mit wöchentlichen Akzeptanztests sowie die inkrementelle Vorgehensweise führen dabei zu einer Entwicklung, die schnell auf Änderungen an Anforderungen reagieren und diese in das Produkt integrieren kann.

Klassische V-Modell-Elemente, wie z. B. eine detaillierte aber auch pflegbare Produktdokumentation, die in der zukünftigen Wartungsphase unverzichtbar sein werden, eine straffe Projektplanung, die laufend fortgeschrieben werden muss, um trotz und gerade wegen der Änderungen ständig aktuell zu sein, und ein effektives Konfigurations- und Versionsmanagement sind andererseits unverzichtbare Bestandteile.

Die Berücksichtigung von Änderungen im Projektverlauf macht ein einerseits

nachvollziehbares und andererseits handhabbares Änderungsmanagement zu einem der stärksten Steuerungsinstrumente des Projektmanagements.

Heutzutage werden insbesondere im eGovernment-Umfeld schnell neue Lösungen verlangt. Um dies zu erreichen sind flexiblere und agilere Vorgehensweisen bei der Softwareentwicklung erforderlich.

Wir möchten die Leserinnen und Leser anregen, uns Ihre Einschätzungen zu den hier vorgestellten Erfahrungen mitzuteilen und uns von eigenen Erfahrungen zu berichten.

Literatur

[And04] Jens Andexer, Hermann Kinder, „Obelix: Die Migration der Daten“, LDVZ-Nachrichten 01/2004

[Col04] Jens Coldewey, „Kooperation statt Misstrauen – Kundenbeziehung in agilen Projekten“, www.objektspektrum.de, OS 05/2004

[Col05] Jens Coldewey, „Projektdämmerung“, www.objektspektrum.de, OS 01/2005

[Eck04] Jutta Eckstein, „Agile Softwareentwicklung im Grossen“, dpunkt.verlag, 2004

[Ger03] Christiane Gernert, „Agiles Projektmanagement – Risikogesteuerte Softwareentwicklung“, Hanser, 2003

[Hag04] Ulrich von Hagen, „Agil und extrem – neue Wege in der Softwareentwicklung“, LDVZ-Nachrichten 02/2004

[Kür02] Petra Kürschner, Ralf Gerson, „Testmanagement und Testautomatisierung im Projekt Obelix“, LDVZ-Nachrichten 01/2002

Dr. Jan Mütter

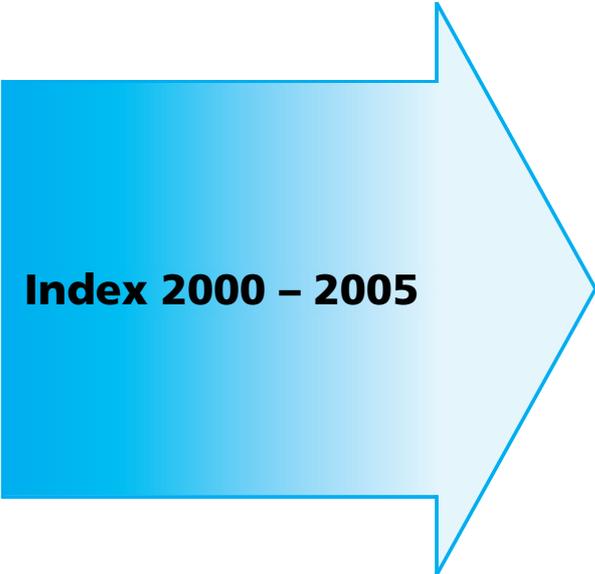
Telefon 0211 9449-5434

E-Mail: jan.muetter@lds.nrw.de

Dipl.-Inform. Ulrich von Hagen

Telefon 0211 9449-6706

E-Mail: ulrich.von-hagen@lds.nrw.de



Index 2000 – 2005

Ausgabe	Schwerpunktthema
1/2005	<p>Einsatz von Open Source Software im LDS NRW</p> <p>Web Service – Technologie der Zukunft?</p> <p>Serverbereitstellung in der Landesdatenverarbeitungszentrale</p> <p>Einrichtung sicherer Verzeichnisse für Arbeitsgruppen mit Windows-Technologien</p> <p>Mono: .Net für Linux</p> <p>WAVE – Eine umfassende Unterstützung für die Veranstaltungsplanung und -durchführung</p> <p>Das neue V-Modell XT</p> <p>Agil und extrem (2) – Praktische Erfahrungen mit agilen Methoden</p>
2/2004	<p>Management moderner IT-Infrastrukturen</p> <p>Barrierefreie Webinformationsangebote</p> <p>Bibliotheksverbund der Landesbehörden</p> <p>OBELIX: Die Migration der Daten</p> <p>IT-Lehrveranstaltungen – einmal anders</p> <p>Agil und extrem – neue Wege in der Softwareentwicklung</p>
1/2004	<p>SPAM – und (k)ein Ende?</p> <p>Open Source Software auf Webservern im LDS NRW</p> <p>Gefahren durch Viren</p> <p>OBELIX und die Systemtechnik</p>
2/2003	<p>SAPOS® – ein modernes Werkzeug zur Positionsbestimmung und Vermessung – realisiert durch Kooperation der Landesbetriebe</p> <p>SPAM – Wenn der Postmann zwei(tausend)mal klingelt</p>

Noch: 2/2003**Zentrale Problemkoordinierung durch das IT-Managementzentrum****ArcGIS-Entwicklungen des Graphikzentrums im LDS NRW
– Automatisch erzeugte Kartenrahmen für ArcGIS 8.x –****Internetzugang der Landesverwaltung NRW****Landesdatenbank im Web – Status und Planung****Datendrehscheibe – Einleiterüberwachung – Abwasser
D-E-A: Stand und Entwicklungsperspektiven****DV-technisches Systemdesign im Projekt Obelix****Wie hältst Du's mit den EJBs?
Webanwendungen mit Java im Überblick****1/2003****GeoServer der Landesverwaltung NRW nimmt Produktion auf****Angebot zentraler GIS-Dienste mit dem GeoServer****Landtagsinformationssystem NRW****Microsofts .Net-Technologie
– wohin geht die Programmierung?****Programmierung mit dem VisualAge Generator im Projekt OBELIX****Webhosting-Dienstleistungen des LDS NRW****1/2002****Einführung des Geoinformationssystems „ArcGIS“
in der Landesverwaltung NRW mit Unterstützung
des Grafikzentrums im LDS NRW****Testmanagement und Testautomatisierung im Projekt Obelix****Contentmanagement in NRW
Ausbau der Webinformationsangebote der Landesverwaltung
und Erleichterung des Zugangs zu Informationen****FÜSYS
Ein Führungsinformationssystem zum strategierorientierten
Management erfolgsrelevanter Projekte in Ministerien
und anderen großen Behörden**

Ausgabe	Schwerpunktthema
Noch: 1/2002	<p>SAN macht Daten schnell Storage Area Network im Windows-LAN des LDS NRW</p> <p>Die Netzbasis des Landesverwaltungsnetzes – gerüstet für die Zukunft</p> <p>'Penguin meets Dinosaur' Linux und Großrechner finden zueinander</p>
2/2001	<p>Das Internet als neues Medium für die Erhebung statistischer Daten</p> <p>Dienstleistungen und Servicekonzepte des LDS NRW zur IT-Infrastruktur</p> <p>Die Erhebung der Abwasser abgabe – DV-Unterstützung einer gesetzlichen Aufgabe</p> <p>GEOSERVER der Landesverwaltung NRW im Intranet und Internet</p> <p>Der Umweltdatenkatalog NRW Transparente Umweltinformationen für Öffentlichkeit, Politik und Planung</p> <p>Kostensenkung am PC-Arbeitsplatz durch Geschäftsprozessanalyse im Desktop Management des LDS NRW</p> <p>Bibliotheksverbund der Landesbehörden NRW</p> <p>Web Based Training im Landesverwaltungsnetz</p>
1/2001	<p>Kommunikation ist alles – Elektronische Post</p> <p>Landtagsinformationssystem Nordrhein-Westfalen</p> <p>Public Key Infrastruktur – Was steckt dahinter?</p> <p>Normen zur Software-Ergonomie</p> <p>IT-Unterstützung im Support – Nutzung eines Trouble-Ticket-Systems</p> <p>„Gut Werkzeug – halbe Arbeit“ Unterstützung der Anwendungsbereitstellung durch Vorgehensmodelle, Methoden und Werkzeuge</p> <p>Obelix – das neue Bezügeverfahren</p>

Ausgabe	Schwerpunktthema
----------------	-------------------------

2/2000	LDS NRW – Internet-Dienstleister der Landesverwaltung Nordrhein-Westfalen
---------------	--

Web-Anwendungen auf dem IBM-Großrechner

Lehrereinstellungsverfahren in Nordrhein-Westfalen

Spracherkennungssoftware – schon einsatzbereit?

**Integriertes Netz- und Systemmanagement
– Selbstzweck oder Wunderwaffe?**

1/2000	Das Büro auf Reisen Mobilität und Datenaktualität – mobile computing Mobile Telefone, Smartphones, Handhelds, PDAs, Notebooks – eine Bestandsaufnahme
---------------	--

**IT-gestützte Vorgangsbearbeitung
in der Landesverwaltung Nordrhein-Westfalen**

**DV-Vorhaben D-E-A
(Datendrehscheibe, Einleiterüberwachung, Abwasser)**

**Zertifizierung des IT-Qualitätsmanagements der LDVZ
nach DIN EN ISO 9001**

Das Windows-NT-Netz im LDS NRW