Web-basierte Bildverteilung im Krankenhaus
Das Java-Projekt *Jive*

Jörg Holstein, Klaus Kleber, Andreas Schröter, Paul Kriener, Thorsten Geisbe,
Dietrich H.W. Grönemeyer*

Institut für Mikrotherapie
Universitätsstr. 142, 44799 Bochum
*Universität Witten/Herdecke, Lehrstuhl für Radiologie und Mikrotherapie
Alfred-Herrhausen-Str. 50, 58455 Witten
Email: holstein@microtherapy.de

Zusammenfassung. Das im folgenden beschriebene Projekt *Jive* konzentriert sich auf die Evaluation der Eignung der Java-Technologie für die Bildverteilung in klinischen Intranets. Ein besonderes Augenmerk wird hierbei auf die Bildverteilung über eine elektronische Patientenakte, auf die Bildverteilung ohne PACS sowie auf den Einsatz von Push- und Pull-Mechanismen gerichtet.

Schlüsselwörter: Bildverteilung, Java, DICOM, Internet/Intranet, Elektronische Patientenakte.

1 Einleitung

Jive ist ein Projekt des Instituts für MikroTherapie der Universität Witten/Herdecke. Es ist zu der Zeit entstanden, als sich die Programmiersprache Java als die Internet-Programmiersprache zu entwickeln begann. Aus der Stellung der Radiologie innerhalb der Krankenhäuser (Leistungsauftrag - Leistungserbringung) ergab sich im Zusammenhang mit der informationstechnischen Entwicklung die Notwendigkeit zur digitalen Bildverteilung, z.B. innerhalb einer elektronische Patientenakte.

Ein primäres Ziel des Projektes war es nun, zu überprüfen, inwieweit sich Java für die Entwicklung eignet, um auch den Anforderungen an die web-basierte Bildverteilung und -bearbeitung innerhalb klinischer Intranets zu gen ügen.

Wesentliche Anforderungen ergaben sich aus den informationstechnischen Randbedingungen. So erfordert beispielsweise die bestehende Basis an Informationssystemen im radiologischen Umfeld (KIS, RIS, PACS) ein hohes Maß an Integrationsfähigkeit für die Bildverteilungskomponenten, die über vorhandene Standardschnittstellen (DICOM, HL7) mit den bestehenden Systemen kommunizieren sollen [1].

Für die Bildbearbeitungskomponenten war gefordert, auch gehobenen Ansprüchen zu genügen und grundlegende Interaktionsmechanismen (Zoom, Grauwertfensterung, etc.) bereitzustellen.

Es wurden weitere Fragestellungen in puncto Technik und Organisation aufgestellt und im Projektverlauf erweitert. So z.B. die Frage, ob sich tatsächlich installa-

tions- und administrationsfreie Clientanwendungen realisieren lassen und wie sich Übertragungs- und Antwortzeiten optimieren lassen.

Ein weiterer Aspekt ist die Umsetzung relevanter Datenschutzkriterien sowie die Erfüllung entsprechender Richtlinien insbesondere vor dem Hintergrund der Nutzung von Internettechnologien (Webbrowser etc.) und des Applet-Konzeptes im Zusammenhang mit verteilten kooperierenden Anwendungen (elektronische Patientenakte).

2 Vorgehensweise

Bei der Konzeption und Implementierung der Bildverteilungs- und der Bildbearbeitungskomponenten wurde in drei Schritten vorgegangen, um konzentriert Teilprobleme herauszuarbeiten und Lösungen dafür zu finden.

Im ersten Schritt ging es darum, die DICOM-Schnittstelle zum PACS (Picture Archiving and Communication System) zu realisieren, eine grundlegende Benutzerverwaltung mit Zugriffsschutzmechanismen zu erstellen sowie das Benutzerinterface für die Bildbearbeitung mit einer entsprechenden Schnittstelle für die Rahmenanwendungen bereitzustellen. Hier sollte insbesondere die Eignung von Java zur interaktiven Bildbearbeitung auf Client-Seite evaluiert werden. Die Anwendungsschwerpunkte waren die elektronische Patientenakte (EPA) sowie das Reviewing [2].

Im zweiten Schritt sollte im Hinblick auf den Anwendungsschwerpunkt "Bildverteilung ohne PACS" die Konzeption und Implementierung eines Kurzzeitarchives (Datenhaltung und -bereitstellung) durchgeführt werden. Wesentliche Aspekte hierbei sind die implizite Zuordnung einer Untersuchung zu einer anfordernden Station bzw. einem Überweiser sowie die Optimierung von Übertragungs- und Antwortzeiten [2].

Für den dritten Schritt stehen weitere logistische Fragestellungen auf dem Plan. Dies betrifft insbesondere Auto-Routing Mechanismen in einer sinnvollen Kombination von Push- und Pull-Bildverteilung. Hierzu werden Erweiterungen der Benutzerverwaltung sowie des Datenschutzkonzeptes durchgeführt. Die fokussierten "Logistikprobleme" entstehen vor allem bei sehr großen oder verteilten Installationen (mit WAN Anbindung) sowie bei der expliziten Bildverteilung zum Überweiser.

3 Ergebnisse

Das Ergebnis der grundlegenden Konzeption ist die Unterteilung der Systemkomponenten auf Basis einer 3-Ebenen-Architektur. Auf der untersten Ebene befinden sich die Datenquellen (KIS, RIS, PACS) mit denen die *Jive Server* (auf mittlerer Ebene) kommunizieren und für die Client-Anwendungen verfügbar machen (Abb. 1).

3.1 Bildverteilung über eine elektronische Patientenakte

Im ersten Schritt des Projektes arbeitete der *Jive Server* als eine reine "Vermittlungsstelle" mit einer primitiven Benutzerverwaltung. Dieser *Jive Server* beinhaltet einen

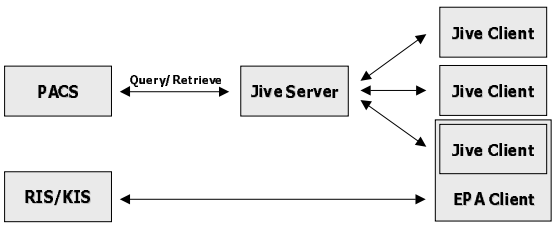

Abb. 1 Kommunikationsschema von *Jive* in der 1. Phase

Webserver, über den die Anwendungsprogramme (Java Applets) von den Client-Systemen abgerufen werden können. Hierdurch werden die Clients immer automatisch mit den aktuellen Programmversionen versorgt.

Nach dem Programmstart meldet sich der Benutzer am *Jive Server* an. Autorisierte Benutzer können dann über den Server ganze Studien oder Serien recherchieren und anschließend vom PACS anfordern. Dazu unterstützt der *Jive Server* die DICOM Query/Retrieve Service Class als Service Class User (SCU). Die Kommunikation zwischen *Jive Server* und *Client* erfolgt über ein proprietäres Protokoll, welches einige Erweiterungen zum DICOM Protokoll, z.B. die Benutzeridentifikation, vorsieht. Die Bilddaten werden bei dieser ersten Lösung nach dem Pull-Prinzip immer erst bei Bedarf übertragen [1].

Die Client Anwendungen, die *Jive Viewer*, sind Java Applets, mit denen die geladenen Bilddaten im vollen Grauwertumfang bearbeitet werden können. Mit dem *Jive Viewer* ist es möglich, so z.B. ganze CT-Serien zu laden, diese in einer Übersichtsanzeige darzustellen und einzelne selektierte Bilder nachzubearbeiten (Fensterung etc.). Die wesentlichen Funktionen, vom Anfordern einer Studie, Serie oder eines Bildes bis hin zum Einstellen der Grauwertfensterung sind über eine Anwendungs-Programmierschnittstelle (API) für eine Rahmenanwendung steuerbar. Dies ermöglicht eine einfache Einbindung der Bildbearbeitungskomponenten in die führende Rahmenanwendung (EPA, RIS, KIS), sie ist aber auch ohne einen solchen Rahmen sehr einfach und flexibel als Review-Station einsetzbar.

Das *Jive* System der ersten Phase wurde in einem mehr-monatigen Testbetrieb im Institut für Mikrotherapie ausgiebig überprüft und optimiert sowie in einer modifizierten Version an der Heinrich-Heine-Universität Düsseldorf zu Lehrzwecken eingesetzt. Die erste Phase ist bereits seit Ende 1998 abgeschlossen [2].

3.2 Bildverteilung ohne PACS

In der zweiten Phase wurde der *Jive Viewer* aufgrund zahlreicher Anregungen und technischer Weiterentwicklungen (Java Swing Klassen) völlig neu konzipiert und implementiert. Dies hat sowohl deutliche Verbesserungen bei der Performance als auch bei der Oberflächengestaltung ermöglicht. So ist die Oberfläche des Viewers nun sowohl optional durch den Benutzer als auch über die Rahmenanwendung konfigurierbar (Erscheinungsbild, Anordnung, Sprache, Darstellungsmatrix, Bildberei-

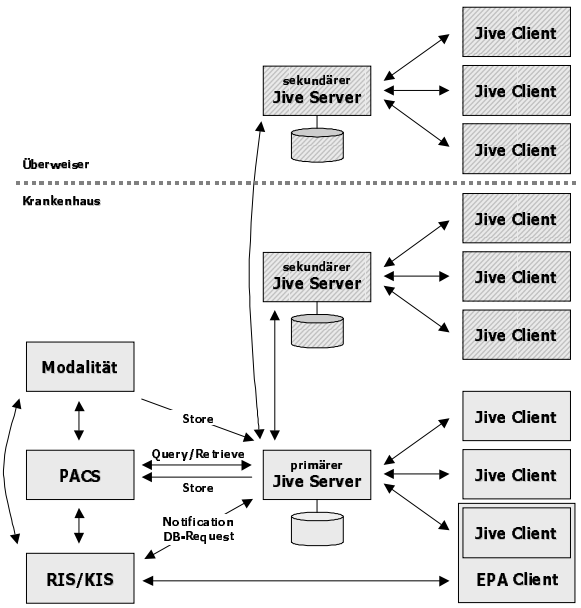

Abb. 2 Kommunikationsschema von *Jive* nach der 2. und 3. Phase

che etc.) und es konnten weitere Funktionen wie Cine, ROI-Zoom und Undo implementiert werden. Außerdem verfügt der Viewer nun auch über eine Befundtextanzeige. Der Viewer steht mittlerweile auch als eine Software-Komponente (Java Bean) zur Verfügung. Dies ermöglicht weitergehende Integrationsmöglichkeiten der Clientanwendung in beliebige Rahmenanwendungen.

Um für den Anwendungsschwerpunkt "Bildverteilung ohne PACS" gerüstet zu sein, waren auch auf Seiten des Servers einige Erweiterungen notwendig. Der Server soll direkt von einer Modalität Bilddaten empfangen können, diese optional komprimieren und kurzzeitig speichern können sowie ggf. ein RIS über den Eingang einer neuen Serie/Studie informieren können.

Die Modalität (oder das PACS) schickt dem *Jive Server* die entsprechende Bildfolge (C-Store). Der Server speichert die eingegangenen und ggf. komprimierten Daten in der eigenen Datenbank und informiert das RIS über den Eingang (Notification). Mit der Datenbank können durchaus Datenmengen in der Größenordnung verfügbarer RAID-Systeme (ca. 150 GB) verwaltet und schnell abgerufen werden (Abb. 2).

Die Implementierung der zweiten Phase ist abgeschlossen und das aktuelle System wird derzeit im Klinikeinsatz erprobt [2].

3.3 Push- und Pull-Bildverteilung

In dieser derzeit laufenden dritten Projektphase sollen die angesprochenen Logistikprobleme dann mit einer Server-Server-Kommunikation gelöst werden. Unterschieden werden hierbei sekundäre und primäre *Jive Server*, die anhand ausgewerteter Strukturdaten die Studien bzw. Serien nach dem Push-Prinzip von dem primären *Jive Server* geschickt bekommen. Hierzu werden Auto-Routing-Regeln benutzt, die in einer konfigurierbaren Regelbasis abgelegt sind.

Mit den sekundären *Jive Servern* können so Stations- oder Abteilungsserver aufgebaut werden. Die *Jive Clients* können die Bilddaten dann nach wie vor nach dem Pull-Prinzip von einem sekundären Server abholen.

Für externe Überweiser können die Bilddaten von einem autorisierten Arzt explizit auf einen speziellen sekundären *Jive Server* (lokal oder entfernt) verschickt werden, von wo aus diese sich die Daten bei Bedarf abholen können (Abb. 2).
Diese neuen Möglichkeiten werfen einige Fragestellungen in puncto Datenschutz (Zugriffskontrolle und -dokumentation, Autorisierung, gesicherte Übermittlung, etc.) auf, die an dieser Stelle allerdings nicht weiter diskutiert werden sollen. Es sei hierzu auf andere Arbeiten mit entsprechenden Schwerpunkten (z.B. Einsatz von Smartcards etc.) verwiesen [3].

4 Schlußfolgerungen und Diskussion

Die bisherigen Erfahrungen haben gezeigt, daß die Internettechnologien und allen voran Java sehr gut für die Lösung der vorgestellten Aufgaben geeignet sind. Die Programmiersprache ist mittlerweile deutlich ausgereifter als noch zu Beginn des Projektes (Plattformunabhängigkeit, Geschwindigkeit und Stabilität), wenngleich die Betriebssystem- und Browser-Hersteller nicht so recht mit der Unterstützung aktueller Versionen mitkommen. Hierdurch ist die Installationsfreiheit für die Clientanwendungen noch eingeschränkt, da mit unter ein entsprechendes Java-Plugin für den Browser benötigt wird. Mittelfristig sollte dies jedoch kein Problem darstellen, da die wesentlichen Hersteller für verschiedene Plattformen ihre Unterstützung angekündigt haben.

Wir haben umfangreiche Tests mit sehr großen Studien (bis zu 200 12-Bit Bilder, 512er-Matrix) und mit großen Bildern (10MB) durchgeführt und konnten bzgl. der Performance und Stabilität auf Client-Seite sehr gute Ergebnisse erzielen; auch die ersten Ergebnisse mit Java auf Server-Seite sind recht vielversprechend.

Als problematisch hat sich erwiesen, daß im DICOM Standard nicht die Markierung der befundrelevanten Bilder vorgesehen ist. Hierdurch wäre es ohne zusätzliche Redundanz möglich, automatisch nur diese ausgewählten Bilder zu übertragen und dadurch die Übertragungszeiten deutlich zu reduzieren [1].

Es wäre daher wünschenswert, daß sich die Softcopy Presentation States (Supplement 33) in der klinischen Routine durchsetzen würden, da mit ihnen auf die relevanten Bilder verwiesen werden kann. Außerdem könnten mit ihnen wichtige

Informationen für den Auftraggeber (Annotationen, Shutter etc.) auf den Bildern angebracht werden und von den *Jive Clients* optional angezeigt werden [4].

5 Literatur

1. NEMA Standards Publications, Digital Imaging and Communications in Medicine (DICOM), National Electrical Manufacturers Association, Rosslyn VA, 1998. http://www.nema.org/nema/dicom/
2. Holstein J, Kleber K, Schröter A, Kriener P, Grönemeyer DHW: Jive – Java based image distribution in medical intranets. In: Lemke HU (Hrsg.) Computer Assisted Radiology and Surgery. Elsevier Science, 1999.
3. Rau WS, Marquardt K, Tolxdorff T (Hrsg.): Tagungsunterlagen des 6. interdisziplinären KIS-RIS-PACS Workshops in Rauischholzhausen, 1999. http://www.uni-giessen.de/kis-ris-pacs/
4. Eichelberg M, Riesmeier J, Kleber K, Holstein J, Grönemeyer DHW, Jensch P: DICOM Presentation States – Ein neuer Dienst für die digitale Bildverteilung und Softcopy-Befundung. In: Bildverarbeitung für die Medizin 2000 – Algorithmen, Systeme, Anwendungen. Springer-Verlag, Berlin, 2000.